Variational Fluid Motion Estimation with Physical Priors

Inauguraldissertation
zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften
der Universität Mannheim

vorgelegt von
Dipl. Inf.
Paul Ruhnau
aus Neustadt a.d. Weinstraße

Mannheim, 2006

Dekan:	Professor Dr. Matthias Krause, Universität Mannheim
Referent:	Professor Dr. Christoph Schnörr, Universität Mannheim
Korreferent:	Professor Dr. Bernd Jähne, Universität Heidelberg
Tag der mündlichen Prüfung:	31. Mai 2007

Abstract

In this thesis, techniques for Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV) are developed that are based on variational methods. The basic idea is not to estimate displacement vectors locally and individually, but to estimate vector fields as a whole by minimizing a suitable functional defined over the entire image domain (which may be 2D or 3D and may also include the temporal dimension). Such functionals typically comprise two terms: a data-term measuring how well two images of a sequence match as a function of the vector field to be estimated, and a regularization term that brings prior knowledge into the energy functional.

Our starting point are methods that were originally developed in the field of computer vision and that we modify for the purpose of PIV. These methods are based on the so-called optical flow: Optical flow denotes the estimated velocity vector inferred by a relative motion of camera and image scene and is based on the assumption of gray value conservation (i.e. the total derivative of the image gray value over time is zero). A regularization term (that demands e.g. smoothness of the velocity field, or of its divergence and rotation) renders the system mathematically well-posed. Experimental evaluation shows that this type of variational approach is able to outperform standard cross-correlation methods.

In order to develop a variational method for PTV, we replace the continuous data term of variational approaches to PIV with a discrete non-differentiable particle matching term. This raises the problem of minimizing such data terms together with continuous regularization terms. We accomplish this with an advanced mathematical method, which guarantees convergence to a local minimum of such a non-convex variational approach to PTV. With this novel variational approach (there has been no previous work on modeling PTV methods with global variational approaches), we achieve results for image pairs and sequences in two and three dimensions that outperform the relaxation methods that are traditionally used for particle tracking.

The key advantage of our variational particle image velocimetry methods, is the chance to include prior knowledge in a natural way. In the fluid environments that we are considering in this thesis, it is especially attractive to use priors that can be motivated from a physical point of view. Firstly, we present a method that only allows flow fields that satisfy the Stokes equation. The latter equation includes control variables that allow to control the optical flow so as to fit the apparent velocities of particles in a given image pair. Secondly, we present a variational approach to motion estimation of instationary fluid flows. This approach extends the prior method along two directions: (i) The full incompressible Navier-Stokes equation is employed in order to obtain a physically consistent regularization which does not suppress turbulent flow variations. (ii) Regularization along the time-axis is employed as well, but formulated in a receding horizon manner contrary to previous approaches to spatio-temporal regularization.

Ground-truth evaluations for simulated turbulent flows demonstrate that the accuracy of both types of physically plausible regularization compares favorably with advanced cross-correlation approaches. Furthermore, the *direct* estimation of, e.g., pressure or vorticity becomes possible.

Zusammenfassung

In dieser Arbeit werden Techniken für Particle Image Velocimetry (PIV) und Particle Tracking Velocimetry (PTV) entwickelt, die auf Variationsansätzen basieren. Die grundlegende Idee dabei ist, Bewegungsfelder nicht lokal zu schätzen, sondern global durch die Minimierung eines geeigneten Energiefunktionals, das im gesamten Bildbereich definiert ist, zu bestimmen. Dieser Bildbereich kann 2D oder 3D sein und auch die zeitliche Dimension mit einschließen. Solche Funktionale bestehen typischerweise aus zwei Termen: Der Datenterm misst, wie gut zwei Bilder eines zu berechnenden Vektorfeldes aufeinander abgebildet werden, und der Regularisierungsterm läßt Vorwissen in das Funktional einfließen.

Den Ausgangspunkt unserer Arbeit stellen Methoden dar, die ursprünglich im Bereich der Computer Vision entwickelt wurden und die wir für PIV modifizieren. Diese Methoden basieren auf dem sogenannten optischen Fluss – der geschätzten Geschwindigkeitsverteilung, die sich durch eine relative Bewegung von Kamera und Szene ergibt. Der optische Fluss basiert darauf, dass man Grauwerterhaltung annimmt (die Materialableitung des Grauwerts der Bildfunktion über die Zeit soll null sein). Ein Regularisierungsterm (beispielsweise die Annahme, dass das Vektorfeld oder dessen Divergenz oder Rotation glatt ist) macht das Problem mathematisch gutgestellt. Experimentelle Untersuchungen zeigen, dass unser Variationsansatz in der Lage ist, Standard-Kreuzkorrelationsverfahren zu übertreffen.

Um unsere Variationsansätze auch für PTV anzupassen, ersetzen wir den kontinuierlichen Datenterm durch einen diskreten, nicht differenzierbaren Matching-Term. Dies führt zu der Frage, wie sich solche Datenterme zusammen mit kontinuierlichen Regularisierungstermen minimieren lassen. Wir erreichen das Ziel, in dem wir eine fortschrittliche mathematische Methode verwenden, die die Konvergenz eines solchen nicht-konvexen Funktionals zu einem lokalen Optimum garantiert. Mit diesem neuen Ansatz erzielen wir bessere Ergebnisse für Bildpaare und Bildsequenzen in zwei und drei Dimensionen als mit relaxationsbasierten Ansätzen, die normalerweise für solche Aufgabenstellungen verwendet werden.

Der Hauptvorteil von Variationsansätzen für PIV ist allerdings die Möglichkeit, Vorwissen auf eine natürliche Art und Weise einzubringen. Da wir strömungsmechanische Bilder analysieren, bietet sich natürlich vor allem die Verwendung von strömungsphysikalischem Vorwissen an. Wir verwenden zwei verschiedene Arten von physikalischem Vorwissen: Zunächst präsentieren wir eine Methode, die nur Vektorfelder zulässt, die die Stokes-Gleichung erfüllen. Dazu führen wir Kontrollvariablen ein, die den optischen Fluss so kontrollieren, dass er den sich bewegenden Partikelteilchen folgt. Wir erweitern diese Methode schließlich in zwei Richtungen: Zum einen verwenden wir die vollständigen Navier-Stokes Gleichungen, um physikalisch konsistent zu regularisieren und turbulente Geschwindigkeitsschwankungen nicht zu unterdrücken. Zum anderen regularisieren wir zusätzlich entlang der Zeitachse, allerdings – im Gegensatz zu früheren örtlich-zeitlichen Regularisierungstermen – mit einem “receding horizon” Verfahren.

Experimente zeigen, dass beide Arten von physikalisch konsistenter Regularisierung in der Lage sind, die Genauigkeit von modernen korrelationsbasierten PIV-Verfahren zu übertreffen. Zusätzlich ist eine *direkte* Bestimmung z.B. von Druck oder von Wirbelstärke möglich.

Acknowledgments

First of all, I would like to thank Prof. Christoph Schnörr for supervising my dissertation and for giving me the opportunity to work in his group. He introduced me to the exciting field of computer vision and provided me with good counsel as I worked on this thesis. His enthusiasm for this research field has always been very motivating. Moreover, I am grateful to Prof. Bernd Jähne for serving as an external referee of this thesis.

The working atmosphere at the CVGPR group has always been pleasant and I would like to thank all members of the group for the inspiring and enjoyable environment and for the unforgettable trips and events. In particular, I want to thank Annette Stahl who read preliminary versions of this thesis and gave many valuable comments for improvements. I also want to thank Florian Becker, Yuan Jing, Timo Kohlberger and Annette Stahl from the CVGPR motion group for the many inspiring discussions that helped to find solutions to seemingly insoluble problems.

Furthermore, the financial support of the Deutsche Forschungsgemeinschaft (DFG) within the priority programme "Bildgebende Messverfahren in der Strömungsmechanik" is gratefully acknowledged. I want to thank all the co-workers within this priority programme that provided me with good counsel as well as with expert knowledge about fluid mechanics. Especially, I am grateful to Rainer Hain and Christian Kähler (TU Braunschweig), Karsten Roetmann and Jochen Scholz (LLG Göttingen), Sebastian Burgmann (RWTH Aachen), Markus Jehle (Universität Heidelberg) and Torsten Putze (TU Dresden) for making available image material and for their support and advice.

Close collaboration with the researchers of the EU project "Fluid Image Analysis and Description" helped me decisively with my work. I am especially grateful to Johan Carlier and Dominique Heitz from Cemagref, and to Etienne Mémin from INRIA Rennes for providing expert knowledge as well as image material.

Finally, I am very grateful to my parents Margot and Claus Ruhnau for their persistent and motivating support. Margot did a great job proof-reading. Thank you very much!

Last but not least I want to thank my girlfriend Eva Schinko for her tireless moral support and her patience, which helped me especially during the last busy months of thesis writing.

Contents

List of Figures

1. Introduction

1.1. Motivation

There are many technical areas of application where it is necessary to measure flow in liquids. In a motor, e.g., one may want to measure how well gas inserted through nozzles, disperses in the cylinder. An even distribution is very important as it guarantees complete combustion and thus a better performance of the motor.

The shape of the combustion space, the position and placement of the nozzles as well as the velocity with which the liquid is brought into the experimental space must be selected to effect a quick introduction of equal distribution.

It is very difficult, however, to compute such complex flow fields in liquids. This is why one may want to consider methods of image processing: One chooses an experimental setup, thus modeling the system in which the motion of a liquid is to be measured. Particles are brought into this liquid (the so-called "seeding") and the flow is photographed with a high-speed camera. Then the distance the particles have traveled in the flow is measured. As the time interval is known, one can deduct the speed of the flow. Good illumination is important so that all the particles will be visible in each picture recorded by the camera. Furthermore, the interval between two consecutive pictures should be as short as possible.

The main advantage of this method is that it is non-intrusive and that instantaneous velocity fields are obtained. The information may then be used to rate the photographed system and to enhance it.

Usually, cross-correlation methods are used to analyze the recorded image pairs/sequences. While these methods will yield good results in most scenarios they are subject to some fundamental limitations (which are often mitigated by certain add-ons and post-processing procedures, cf. sec. 2.2.1). In this thesis, we will show how to incorporate prior knowledge about the flow in a mathematically sound way by using variational methods.

Experimental evaluation confirms that our type of variational approach is able to outperform standard cross-correlation methods in the following aspects:

- **Resolution:** *Dense* (i.e. one vector per pixel) velocity fields are reconstructed. The size of the interrogation areas does not limit the resolution.
- **Accuracy:** Large velocity gradients are admissible. (No assumption of a negligible motion field variation inside interrogation areas.)
- **Spatial context:** Prior knowledge about spatial flow structures can be exploited during

estimation. Ambiguities are resolved by use of neighborhood information. No need for post-processing.

1.2. Related Work

As Particle Image Velocimetry (PIV) is an industrially very relevant research topic, there exists a vast amount of literature on many methods to analyze particle image pairs and sequences. Furthermore, different commercial software packages for the analysis of PIV image pairs and sequences are available. Nearly all available methods can be devided into one of the following three categories:

(i) Cross Correlation Methods Most methods for analyzing PIV data can be seen as extensions to cross-correlation approaches. A complete survey of these methods would be beyond the scope of this manuscript – a brief review dealing predominantly with the fundamental limitations of cross-correlation approaches will be given in section 2.2.1.

(ii) Particle Tracking Methods While most PIV methods that operate on gray value images are based on cross-correlation, literature clearly presents more distinct particle tracking approaches. Most of these PTV approaches have two steps in common: First the individual particles are extracted from the gray value structure of the image and then the correspondence problem (as to which particle in the first frame corresponds to which particle in the second frame) is solved. In sec. 2.2.2 we will go into different PTV methods.

(iii) Local Optical Flow Methods In the computer vision community, optical flow methods are much more common than cross-correlation-based approaches. Local approaches for optical flow estimation were introduced in the early 1980s by Lucas and Kanade [LK81]. In the last few years these methods (which were originally developed for general motion estimation tasks) have been successfully applied to PIV scenarios. We will sketch local optical flow methods in section 3.3.

(iv) Variational Optical Flow Methods Variational methods for motion analysis go back to the early 1980s [HS81] and were originally developed for more general motion estimation tasks (motion in traffic scenes, robot vision, ...). Since then, there has been a great deal of research on different methods for the recovery of optical flow in different scenarios (e.g. [BFB94, BB95]). This also led to the development of variational methods for the analysis of meteorological flows and fluid flows [WALL97, BHY00, CMP02]. Note that these methods (which we will describe in sec. 4.1.1) were developed for so-called "passive scalar"-scenarios in which the image gray values observe the conservation of mass just as does the fluid density.[1] For imagery obtained by the typical PIV method, these approaches are not adequate. Variational methods that are suited for PIV data have been proposed only very recently [KMS03, CHA+05, YRMS05]. We will describe these methods in section 5.1.

[1] In this manuscript we concentrate on developing methods for the analysis of *particle* image pairs and sequences. At some points (cf. sec. 4.1.4, 5.3.3, and 5.4.3), however, we will show that our methods are able to outperform standard cross-correlation methods' performance for passive scalar image pairs and sequences.

The approaches that we will present in this thesis can be classified into the categories (ii) and (iv). In comparison to the above approaches, the main points of our work are:

- We present variational methods for both PIV and PTV scenarios. In both cases we adapt the data term in a way to fit the individual structure of PIV imagery.
- We do not only adapt the data term to the individual requirements of PIV data but we also use the variational framework in order to include physically motivated prior knowledge into the regularization term.

In this manuscript, we will compare our approaches with the different competing methods whenever this is possible (sec. 4.1.4, 4.2.3, 5.1.4, 5.3.3, 5.4.3). This comparison is performed in order to give the reader an idea about why which methods yield good/bad results on which image data. We believe that reliable conclusions about the advantages and disadvantages of the individual methods are crucial for further development of the individual algorithms.

1.3. Contribution

Our main research interest is to develop new techniques for PIV and PTV based on variational methods. The basic idea is not to estimate displacement vectors locally and individually, but to estimate vector fields each as a whole by minimizing a suitable functional defined over the entire image domain (which may be 2D or 3D and may also include the temporal dimension). Such functionals typically comprise two terms: a data-term measuring how well two images of a sequence match as a function of the vector field to be estimated, and a regularization term that brings physically motivated prior knowledge into the energy functional.

(i) Variational Particle Image Velocimetry [RKNS05, YRMS05, RKNS04] Our starting point are methods that were originally developed in the field of computer vision and that we modify for the purpose of PIV. These methods are based on the so-called *optical flow*: It denotes the estimated velocity vector inferred by a relative motion of camera and image scene and is based on the assumption of gray value conservation (i.e. the total derivative of the image gray value over time is zero). A regularization term (that demands e.g. smoothness of the velocity field or of its divergence and rotation) renders the system mathematically well-posed. Experimental evaluation shows that this type of variational approach is able to outperform standard cross-correlation methods. In chapter 3, we will review the pioneering variational optical flow approach by Horn&Schunck [HS81] and adapt it in chapter 4 to the special requirements of PIV.

(ii) Variational Particle Tracking Velocimetry [RGS05b, RGS05a] Particle Tracking algorithms are becoming more and more popular as they are capable of yielding higher resolution velocity fields (i.e. one vector for every particle image). Furthermore they can be easily supported and combined with 3D stereo reconstruction, leading to high-resolution 3D3C vector fields. We generalize the class of variational approaches (described in (i)) to Particle *Tracking* Velocimetry. To this end, we replace the *continuous* data term of variational approaches to PIV with a *discrete* non-differentiable particle matching term from PTV. This raises the problem of minimizing such data terms together with *continuous* regularization terms. We accomplish this with an advanced mathematical method, which guarantees convergence to a local minimum

of such a non-convex variational approach to PTV. With this novel variational approach (there has been no previous work on modeling PTV methods with global variational approaches), we achieve results for image pairs and sequences in two and three dimensions that outperform the relaxation methods that are traditionally used for particle tracking. We describe our variational particle tracking approach in sec. 4.2.

(iii) Physical Priors [RS06b, RS06a, RS06c, RSS06a, RSS06b] The key advantage of our variational particle image velocimetry methods, is the chance to include prior knowledge in a natural way. Note that the only prior knowledge that we used in (i) is the smoothness of the velocity field (and divergence and curl, resp.).

In sec. 5.3, we present an approach to motion estimation between image pairs based on optical flow estimation subject to *physical* constraints. Admissible flow fields are restricted to vector fields satisfying the Stokes equation. The latter equation includes control variables that allow to control the optical flow so as to fit to the apparent velocities of particles in a given image pair. We show that when the real unknown flow observed through image measurements conforms to the physical assumption underlying the Stokes equation, the control variables allow for a physical interpretation in terms of pressure distribution and forces acting on the fluid. Although this physical interpretation is lost if the assumptions do not hold, our approach still allows for reliably estimating more general and highly non-rigid flows from image pairs and is able to outperform cross-correlation-based techniques.

In sec. 5.4, we present a variational approach to motion estimation of *instationary* fluid flows. This approach extends the prior method along two directions: (i) The full incompressible Navier-Stokes equation is employed in order to obtain a physically consistent regularization which does not suppress turbulent flow variations. (ii) Regularization along the time-axis is employed as well, but formulated in a receding-horizon manner, contrary to previous approaches to spatio-temporal regularization. This allows for a recursive on-line (non-batch) implementation of our estimation framework.
Ground-truth evaluations for simulated turbulent flows demonstrate that due to imposing both physical consistency and temporal coherency, the accuracy of flow estimation will compare favorably even with advanced cross-correlation approaches and optical flow approaches based on higher-order div-curl regularization.

1.4. Organization

Chapter 2 gives a short overview of PIV recording techniques and it additionally sketches the standard methods that are usually applied to analyze PIV imagery. We point out the limitations of the individual methods, thus motivating the use of variational methods in the subsequent chapters.

Chapter 3 gives an introduction to variational approaches in general and then to variational approaches for motion estimation. We also go into the discretization of the arising partial differential equations.

In chapter 4, we adapt the data term of our prototypical variational approach of chapter 3 to the specific demands of PIV data. This yields on the one hand a variational approach for PIV (sec. 4.1) and a variational approach for particle tracking on the other hand (sec. 4.2). In the corresponding experimental sections we show comparisons with standard approaches for

velocity estimation from PIV images.

In chapter 5, we address the question of how to adapt the prototypical regularization term from chapter 3 to the specific demands of PIV velocity fields. We collect different possibilities of minimizing variational motion estimation functionals that incorporate higher order regularization, and show in the experimental section how these improved regularization terms help to estimate more accurate velocity fields. In sec. 5.2, we discuss a possible physical interpretation of the standard regularization terms introduced in sec. 3.4 and 5.1.2. We bring forward the argument that even higher order regularization is physically not really sound. Consequently, we turn towards physically more plausible regularizers in sections 5.3 and 5.4. In sec. 5.3, we use the linearized steady version of the Navier-Stokes equations as a first (simple) physically plausible regularizer. We show that if the flow is actually governed by this Stokes equation, we are not only able to estimate reliable velocity distributions but we can also give pressure and force estimates. Though this interpretation will not be valid if the flow is not governed by the Stokes equation, we show that we can still estimate very reliable velocity fields. In sec. 5.4, we expand the approach to the analysis of whole image sequences. Instead of the (linearized) Stokes equation, we use the vorticity transport equation as prior knowledge. This equation contains the full Navier-Stokes equations and is therefore also valid in turbulent scenarios. Furthermore, it allows the incorporation of temporal coherency in the estimation process.

We conclude in chapter 6 by summarizing our work and indicating open problems and possible extensions.

Appendix A contains the mathematical basis that is needed to discretize and numerically solve the elliptic systems that we encounter throughout the whole manuscript.

Appendix B forms the extension of appendix A to saddle-point problems that we encounter in chapter 5.

In appendix C, we outline the basic equations of fluid mechanics and introduce some relevant special cases and simplifications.

In appendix D, we finally sketch the discretization of the vorticity transport equation that we use as a physically plausible prior in sec. 5.4.

2. Particle Image Velocimetry

This chapter is to serve as a short introduction into what is called “Particle Image Velocimetry” (PIV). The term includes on the one hand the technical background of fluid image capturing (with high-speed camera systems) and on the other hand the algorithmic part of (digital) image sequence analysis. Note that our review is far from being complete – it is rather a collection of different aspects/techniques which we will come back to in the following chapters. For a complete overview on PIV, the reader is referred to [RWK01].
In order to understand the very specific nature of captured fluid image sequences, it is crucial to get an idea of how the images are obtained. Section 2.1 sketches the technical background of the recording process. We start with traditional two-dimensional PIV and subsequently outline generalizations of these techniques to 3D.

2.1. Recording Techniques

Particle Image Velocimetry is an optical method that is used to measure velocities (and other derived quantities) in fluids. As fluids are commonly non-textured, there is a need to add a texture to the fluid so that fluid motion can be perceived at all. For this purpose, tracer particles are usually added to the flow. There is a vast literature on tracer particles of different materials and sizes (for different experimental setups); for a survey we refer to [Mel97].[1]

2.1.1. Traditional 2D PIV Recording

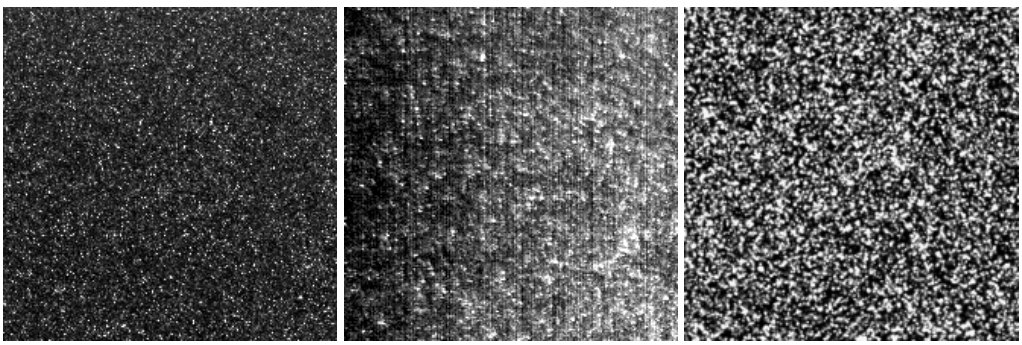

Figure 2.1.: Typical PIV images when using different tracers.

Whole velocity fields are to be measured by taking two (or more) images of the flows, one shortly after the other, and calculating the distance the individual particles have traveled within

[1] Note that there are also different ways to add texture to the fluids. Molecular Tagging Velocimetry (MTV), e.g., is a specific form of velocimetry in which laser beams “write” structures into the fluid. These structures then move with the fluid. Accordingly, PIV image evaluation techniques are applicable in these scenarios as well.

this time. If we assume that the seeded particles have actually followed the motion of the fluid

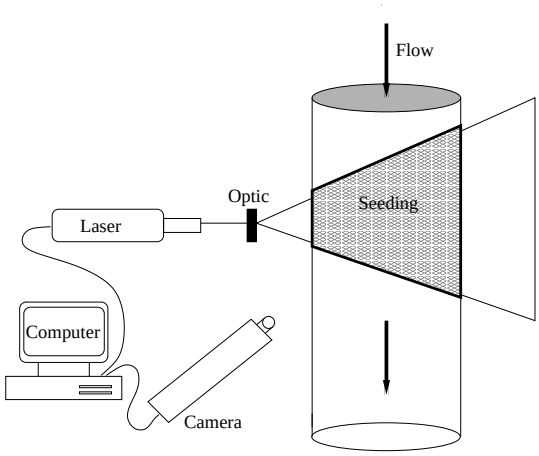

Figure 2.2.: System components for PIV

(an assumption that usually holds true if the added tracers are properly chosen), then the velocity of the fluid can be calculated from the known time difference and the measured displacement. Note that acceleration information cannot (at least not without adding additional constraints) be obtained from the analysis of image pairs only. To avoid blurred images when the flow is fast, laser pulses are used (usually Nd:YAG). As they are only 6-10 ns long, they are capable of freezing any motion. The use of laser pulses has a second advantage: Only laser light can be focused (usually using a cylindrical lens) into a light sheet thin enough so that just the particles on the one plane are imaged. A special CCD camera has to be used. It must be able to store the first image fast enough to be ready for the second exposure. Thus, the "dead" time – when the camera is "blind" between two images – can be reduced to 200 ns [LaV01]. Traditional PIV camera systems allow the capturing of image pairs only.[2]

The main advantage of Particle Image Velocimetry (in contrast to other techniques, like hot-wire anemometry) is that it is *non-intrusive*[3]. This allows the measurement of velocity also in scenarios where probes would distort the velocity field (e.g. high-speed boundary-layer flows). In contrast to hot-wire techniques, it is furthermore possible to measure the flow velocity in a whole cross-section of the fluid in parallel (*whole field technique*). Huge amounts of flow data can be obtained in a comparatively short period of time. This allows for a statistical evaluation of flow field properties over time.

There are, however, also disadvantages: Tracers have to be seeded into the fluid; this seeding is often not possible in real applications (e.g. due to temperature). Furthermore, it is often problematic to position the tracers in the very locations where the flow is to be measured. Experimental conditions sometimes forbid the rather sophisticated positioning of the different PIV components.
Furthermore, the described experimental setup is only capable of yielding 2D velocity fields. If

[2] Note that this limitation is overcome by some recent (expensive) high-speed camera systems.
[3] It has been shown that properly chosen tracer particles generally cause only negligible distortion of the velocity field [Mel97].

a tracer particle leaves the illuminated plane (out-of-plane motion), the particle fades or even disappears. On the other hand, particles can enter the illuminated plane from one frame to the next frame. Both scenarios will lead to motion estimation errors – no matter which algorithm is used for image analysis. To summarize, only 2D motion (i.e. the 2D projection of a 3D motion) can be captured. This is a decisive drawback, as fluid motion is intrinsically 3D. In the next section we will see, however, how the principles of PIV can be generalized to 3D.

2.1.2. Generalization to 3D

The most straight-forward method to obtain information about the out-of-plane component is the use of one or two additional cameras (or alternatively a system of mirrors placed in front of a single camera). For a review of stereoscopic PIV, we refer to [Pra00] and references therein. Note that these techniques are capable of yielding all three components (3C) of the velocity. However, only velocity vectors in a 2D slice are recovered. These types of stereoscopic techniques are therefore referred to as *2D3C*. If enough illumination is available, the thickness of the laser light sheet can be expanded. In this way, velocity information inside a 3D cube can be obtained (3D3C)[BAP94, Maa92a, Maa92b].
A different family of approaches (*dual plane PIV*, [RWW+96]) uses a third recording while the laser light sheet is slightly shifted. On the basis of the relative motion from one light sheet to the next, 3D motion can be reconstructed. This technique is expandable to measuring the velocities in a whole volume as well [Brü95, Brü96] if one uses a scanning light-sheet setup.

2.2. Standard Evaluation Methods for Fluid Images

With the knowledge of how the individual images are captured, we want to turn to the actual image processing now. The terms *cross-correlation particle image velocimetry* (CC-PIV) and *particle tracking velocimetry* (PTV) denote established classes of image processing methods for extracting the underlying velocity fields in particle images. CC-PIV methods operate on gray-level images, while PTV approaches determine the flow field by tracking individual tracers [RWK99].

2.2.1. Cross-Correlation PIV

In this section we want to introduce cross-correlation particle image velocimetry (CC-PIV). CC-PIV has become the best-known and most widely used experimental method for flow estimation.

Let $I(x_1, x_2, t)$ denote the gray value recorded in the image plane at location $(x_1, x_2)^\top$ and time t. A basic assumption underlying most approaches to motion estimation (including cross-correlation PIV approaches *and* optical flow approaches) is that I is conserved, that is the change of $I(x_1, x_2, \cdot)$ at location $(x_1, x_2)^\top$ is due to a movement of $I(x_1, x_2, t)$ to the location $(x_1 + u_1\Delta t, x_2 + u_2\Delta t)^\top$ during a time interval Δt:

$$I(x_1 + u_1\Delta t, x_2 + u_2\Delta t, t + \Delta t) = I(x_1, x_2, t) \ .$$

A common approach to estimating $(u_1, u_2)^\top$ at some fixed location $(\overline{x_1}, \overline{x_2})^\top$ on the image grid $(x_1, x_2)^\top = (k_1\Delta x_1, k_2\Delta x_2)^\top,\ k_1, k_2 \in \mathbb{Z}$, is to assume u_1 and u_2 to be constant within a local

spatial area $W(\overline{x_1}, \overline{x_2})$ around $(\overline{x_1}, \overline{x_2})^\top$ and to minimize[4]

$$\sum_{k_1,k_2 \in W(\overline{x_1},\overline{x_2})} \left[I(k_1 + u_1\Delta t, k_2 + u_2\Delta t, t + \Delta t) - I(k_1, k_2, t)\right]^2$$

as a function of u_1 and u_2. Assuming additionally that $\sum_{k_1,k_2 \in W(\overline{x_1},\overline{x_2})} I(k_1, k_2, t)^2$ does not vary with $(\overline{x_1}, \overline{x_2})^\top$[5], the minimizing values of u_1, u_2 maximize the correlation function

$$\phi(x_1, x_2) = \sum_{k_1,k_2 \in W(\overline{x_1},\overline{x_2})} I(k_1 + u_1\Delta t, k_2 + u_2\Delta t, t + \Delta t)\; I(k_1, k_2, t)\;. \tag{2.1}$$

For each choice of a shift (x_1, x_2), the sum of the products of all gray values in a certain neighborhood W (also called *interrogation window*) produces one cross-correlation value $\phi(x_1, x_2)$. Figure 2.3 shows how this is performed in practice: A template of the size of the neighborhood W (here: 4) is extracted from $I(x_1, x_2, t)$ and a sample of the size of the search region is extracted from $I(x_1, x_2, t + \Delta t)$ (here: 8). The template is linearly shifted around the search region. For every integer shift (in our toy example, 25 shifts are possible, with $-2 \leq x_1 \leq 2, -2 \leq x_2 \leq 2$), the corresponding correlation coefficient is computed using (2.1). This yields a whole cross-correlation plane (as indicated in figure 2.4). For those shifts that align the particles of template and sample, the cross-correlation plane will show a maximum.
Standard cross-correlation techniques have some fundamental limitations which can, however, be weakened by certain strategies that we will illuminate in the succeeding sections:

- The process of finding the highest correlation value for every window is time-consuming. The number of multiplications per correlation value increases in proportion to the interrogation window area. Most cross-correlation PIV approaches avoid this costly computation by performing a complex conjugate point-wise multiplication of the two-dimensional Fourier-transformed sub-images (cf. sec. 2.2.1 (i)).

- Due to the statistic nature of cross-correlation PIV, there is a trade-off between interrogation window size and resolution of the recovered velocity estimates. Large windows lead to robust but coarse estimates, small interrogation windows yield higher-resolution estimates, which are, however, error-prone due to noise. In sec. 2.2.1 (ii), we will review some window refinement techniques that are used to recover high-resolution velocity estimates.

- The cross-correlation method recovers only linear shifts. Only one displacement estimate is recovered per interrogation window; the cross-correlation function will peak at the average linear shift of all particles within the interrogation window. However, this peak will be less pronounced in regions with a large velocity gradient. Iterative image deformation techniques (cf. sec. 2.2.1 (iii)) have been suggested by a number of authors.

- Cross-correlation yields only integer value velocity reconstruction. Fractional displacement estimates can be obtained using correlation peak detection and sub-pixel interpolation (sec. 2.2.1 (iv)).

[4] Without loss of generality we take $\Delta x = \Delta y = 1$.
[5] In fact, modern cross-correlation PIV techniques take into account spatial fluctuations in I by normalizing the correlation coefficients.

- Motion estimation is carried out regardless of spatial context. As a consequence, prior knowledge about spatial flow structures is not exploited directly during estimation, and missing motion estimates in image regions, where a correlation analysis yields no reliable estimates, have to be heuristically inferred in a post-processing step (cf. sec. 2.2.1 (v)). Sometimes, physics-based priors (cf. sec. 2.2.1 (vi)) are used which ensure that the resulting velocity field satisfies the continuity equation or the Navier-Stokes equation.

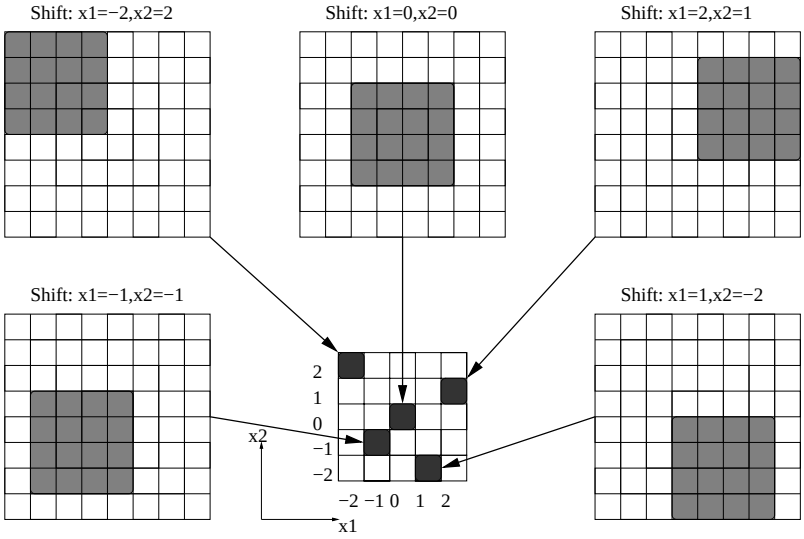

Figure 2.3.: Cross-correlation Overview: Formation of the cross-correlation plane (middle). A 4×4 template is correlated with a 8×8 sample. This yields a 5×5 correlation plane.

(i) Frequency based Correlation

Most cross-correlation PIV approaches avoid the time-consuming calculations of (2.1) by taking advantage of the correlation theorem, which states that the cross-correlation of two functions is equivalent to a complex conjugate point-wise multiplication of their two-dimensional Fourier-transforms. This fact is expressed by the correlation theorem [GW87]. Let $I_1(x_1, x_2) = I(x_1, x_2, t)$ and $I_2(x_1, x_2) = I(x_1, x_2, t + \Delta t)$. Then

$$\phi(x_1, x_2) \Leftrightarrow \hat{I}_1(\xi, \eta)\hat{I}_2^*(\xi, \eta), \tag{2.2}$$

where $\hat{I}_1$ denotes the Fourier transform of the function I_1, and $\hat{I}_2^*$ represents the complex conjugate of the Fourier transform of the function I_2. Transforming the image samples to the Fourier domain, doing the complex-conjugate multiplication (2.2) there and transforming the data back, reduces the complexity[6] of the overall computation from $O(N^4)$ to $O(N^2 log N)$. Note, however, that the finite size of the windows (equivalent to the assumption of the data being periodic) leads to a biasing of the correlation data towards small displacements [RWK01]. This bias can, in turn, be eased if multi-pass techniques are applied (cf. next section).

[6] $N \times N$ is the size of the two input samples.

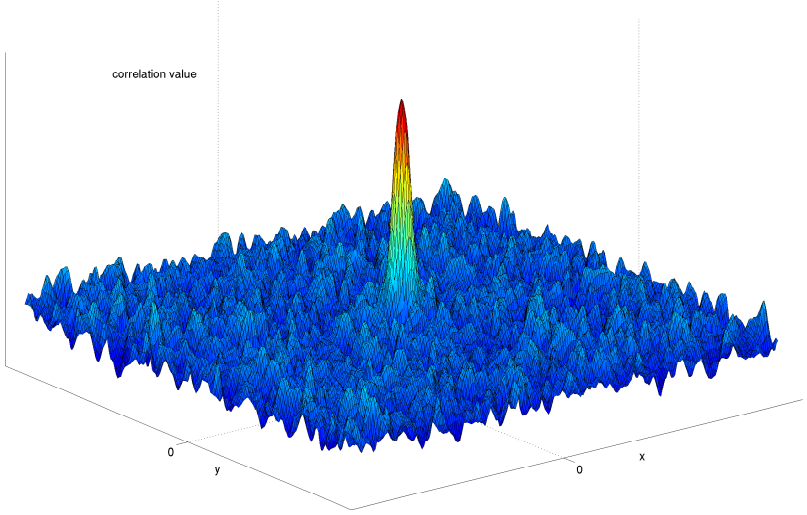

Figure 2.4.: Cross-correlation Plane: If the experimental conditions are good, one exposed correlation peak is generated. Peak detection yields the integer offset $(-1, 4)$.

(ii) Multi-Pass Techniques, Iterative Refinement

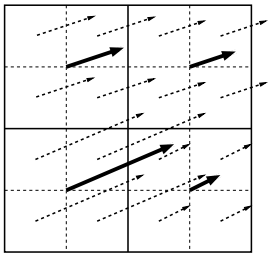

Figure 2.5.: Multi-Pass Computations: Resulting coarse vector fields are interpolated to the fine grid.

It has been shown [WDG97] that by offsetting the correlation windows (according to a previously computed velocity estimate), the number of matched particles increases. Usually, three multi-pass steps are performed.
The resolution and accuracy of the velocity estimates is further increased by iterative refinement techniques [SR99]:

- Compute a first cross-correlation between the two images using a large window size ($N \times N$).
- Scan for outliers and replace by interpolation.

- Halve the size of the interrogation windows ($N \times N \rightarrow N/2 \times N/2$). Project the coarse interrogation result to the new (smaller) windows (cf. fig. 2.5). Offset correlation windows according to this prediction, and perform cross-correlation.
- Repeat items 2 and 3 until the desired resolution is reached.

The choice of the final interrogation window size depends on the particle density. At least 4 particles pairs should be located inside both corresponding interrogation windows.
In contrast to traditional single-pass cross-correlation methods, these multi-pass techniques with iterative refinement allow for the decoupling of maximum in-plane displacement and interrogation window size. This yields superior results in cases where image density and dynamic range in displacements are high. Note, however, that the windows themselves are *not* deformed by this method. Therefore, only linear shifts (i.e. all particles in the window are assumed to travel with the same, constant speed) are taken into account. If this assumption does not hold, the correlation peak (in every iteration) will be less pronounced.

(iii) Iterative Image Deformation Techniques

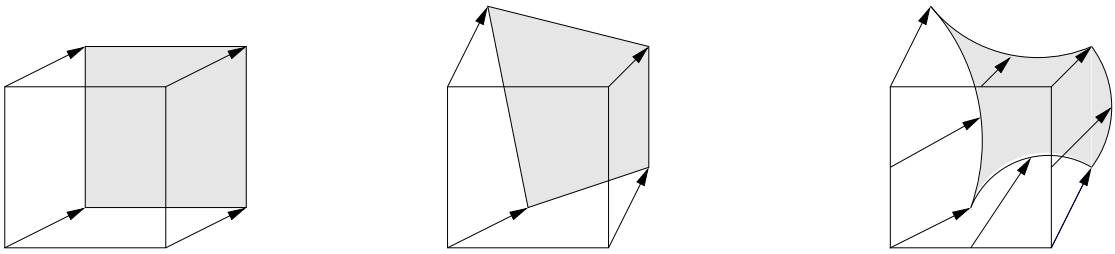

Figure 2.6.: Different orders of image deformation: **Left:** 0th order, **Middle:** 1st order, **Right:** 2nd order

Until now we have only considered algorithms that assume a constant displacement within every correlation window. We have already seen that this assumption yields problems for highly non-rigid flows that exhibit strong velocity gradients within the interrogation windows. To overcome these problems many authors have suggested schemes that iteratively deform the interrogation windows [HFW93, JJDAF95, TD95]. The corresponding displacement distributions vary spatially over the interrogation windows. Scarano [Sca02] classifies the different methods as depending on how many terms of a truncated Taylor series they consider. The Taylor series that is used to estimate the displacement distribution over the finite interrogation region is given by

$$\begin{aligned} u(x_1,x_2) =& u(x_1^0,x_2^0) + \Big(\frac{\partial u}{\partial x_1}\Big)(x_1-x_1^0) + \Big(\frac{\partial u}{\partial x_2}\Big)(x_2-x_2^0) \\ &+ \frac{1}{2}\Big[\Big(\frac{\partial^2 u}{\partial x_1^2}\Big)(x_1-x_1^0)^2 + \Big(\frac{\partial^2 u}{\partial x_1 \partial x_2}\Big)(x_1-x_1^0)(x_2-x_2^0) + \Big(\frac{\partial^2 u}{\partial x_2^2}\Big)(x_2-x_2^0)^2\Big] \\ &+ O(x-x_0)^3 \\ \text{with } x_1 \in \Big[x_1^0 - \frac{1}{2}W, x_1^0 + \frac{1}{2}W\Big] &\text{ and } x_2 \in \Big[x_2^0 - \frac{1}{2}W, x_2^0 + \frac{1}{2}W\Big] \ , \end{aligned}$$

where (x_1^0, x_2^0) denotes the center of the interrogation window W. Note, that the approach described in sec. 2.2.1 (ii) can be classified as a *zero order displacement predictor* as it uses merely $u(x_1, x_2) = u(x_1^0, x_2^0)$ for interpolation. Linear (*first order*) displacement predictors are the most commonly used window deformation techniques (e.g. [HFW93, JJDAF95]): After a first cross-correlation sweep, the (piecewise linear) displacement distribution inside the correlation windows is estimated, using a linear interpolation with respect to the interrogation grid points. Higher-order methods are applied less often, as the number of parameters increases exponentially with the increase in truncation order.

(iv) Correlation Peak Detection and Sub-Pixel Interpolation

Recall that the result of the cross-correlation evaluation is a correlation plane that (hopefully) has one single peak at a certain location. However, correlation values only exist for integer displacements. Nevertheless, it is possible to achieve sub-pixel accuracy by using peak-fitting functions. This is usually done in the following way: One searches for the highest value in the correlation plane. Suppose this value is located at the integer coordinates (i, j). Mainly, three-point estimators are used that use the correlation values at positions (i, j), $(i-1, j)$,$(i+1, j)$,$(i, j-1)$,$(i, j+1)$ in order to fit the peak. There are three well-known interpolation functions: *Peak Centroid, Parabolic Peak Fit* and *Gaussian Peak Fit*.

- **Peak Centroid:** The ratio between first order moment and zeroth order moment is computed. For the x coordinate this yields

$$x_1^0 = \frac{(i-1)\phi(i-1,j) + i\phi(i,j) + (i+1)\phi(i+1,j)}{\phi(i-1,j) + \phi(i,j) + \phi(i+1,j)} ,$$

  and analogue for the x_2 coordinate.

- **Parabolic Peak Fit:** It is more robust to fit the correlation data to some function. If we use three points in every direction, this fit function is parabolic:

$$f(x) = Ax^2 + Bx + C$$

  Straight-forward calculations yield therefore

$$x_1^0 = i + \frac{\phi(i-1,j) - \phi(i+1,j)}{2(\phi(i-1,j) - 2\phi(i,j) + \phi(i+1,j))} ,$$

  and the analogue for the x_2 coordinate.

- **Gaussian Peak Fit:** The most widely used peak fit function is the Gaussian function because the particles themselves can be described very well by Gaussian intensity distributions

$$f(x) = C \exp \frac{-(x^0 - x)^2}{k}.$$

  This leads to the following peak estimator

$$x_1^0 = i + \frac{\ln \phi(i-1,j) - \ln \phi(i+1,j)}{2(\ln \phi(i-1,j) - 2\ln \phi(i,j) + \ln \phi(i+1,j))} ,$$

  and the analogue for the x_2 coordinate.

Note that these interpolation functions are capable of detecting peaks with an accuracy up to 0.05 pixels. However, it has been found independently by many authors, that cross-correlation estimates tend to be biased towards integer displacements. This effect is the stronger, the smaller the particles get. It is clear that the reason for this effect has to lie in the sub-pixel interpolation routine, in fact it is due to interpolation effects. For a detailed discussion of peak locking, we refer to [Wes93].

(v) Post Processing: Data Validation

It is clear that cross-correlation PIV does not take into account spatial context. Therefore, it is likely that cross-correlation analysis will lead to wrong displacement estimates for some interrogation windows. Fortunately, these outliers can be easily detected, as the magnitude and direction of outlier vectors usually differ considerably from those of the surrounding velocity estimates (cf. fig. 2.7). There is vast literature that describes different techniques for outlier de-

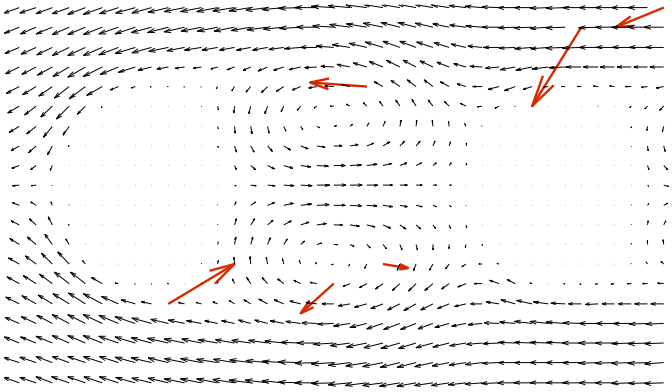

Figure 2.7.: Cross-Correlation analysis yields outliers. These have to be detected and rectified in a post-processing step.

tection and data interpolation (e.g. [RLK93, Wes94, SBB04, WS05]). Most techniques combine the following two points:

- Each velocity vector in the image is compared to its neighbors. The velocity vector is considered an outlier if the absolute difference between its magnitude and the average magnitude of its neighbors is larger than a certain threshold.[7] Alternatively, the divergence is calculated at each position, and the vector is rejected if the local divergence is above a certain level [SR99].

- In a second step, neighboring velocity vectors are used to fill in the missing data. This is usually done by some kind of interpolation scheme.

If multi-pass techniques are applied, it is crucial to validate the data after *every* sweep.

[7] Westerweel [Wes94] has successfully replaced the average by a median.

(vi) Physics-based Priors

Concerning the incorporation of physical constraints for flow estimation through image processing, several interesting approaches have been suggested in the past. Combining PIV and CFD by using cost functions was proposed in [KYKI98, ONI97, AD05]. They can be understood as special *post*-processing methods: By correcting cross-correlation velocity estimates so that they approximately conform to the continuity equation or the whole incompressible Navier-Stokes equations, outliers are reliably detected and rectified.

More recently, physics-based non-linear dynamic models [OSN00] have been introduced to PIV. The velocity is again obtained by minimizing a measure which consists of the residues of the Navier-Stokes equation, the continuity equation, and the difference between estimated and observed image data. The resulting non-linear optimization system is solved, using methods from evolutionary programming [Mic94]. This procedure is repeated until the difference between the observed and the estimated image is sufficiently small. This method allows a reliable estimation of velocity fields and pressure estimates. One may criticize, however, that little insight can be gained from the viewpoint of optimization.

Conclusion

We have derived the standard algorithm for cross-correlation particle image velocimetry and have summarized several extensions. We have seen that the standard algorithm has some limitations that can be eased by iterative schemes and post-processing steps. While the standard cross-correlation approach is very simple, the whole process becomes extremely complex. If errors occur within this process, it is often unclear at what step they were caused. Note that the error rates of up-to-date image processing methods for PIV are far below 1 pixel.[8] This means that the overall error is mainly caused by the peak-fitting function that is used. This continuous function (e.g. Gaussian) is fitted to the input data (discrete correlation coefficients) to achieve sub-pixel accuracy. The interaction between cross-correlation and peak-fitting function is very complex and still topic of vast research (cf. [CK05] and references therein). A second drawback lies in the fact that cross-correlation PIV relies on very specific input data: particle images of a certain size. Its application to other kinds of data (e.g. produced by Molecular Tagging Velocimetry) is problematic as correlation relies on the images' high-frequency components.

Later in this thesis, we will introduce variational motion estimation techniques that provide intrinsically sub-pixel accuracy (no need for peak-fitting), data validation (with corresponding validity constraints that can be provided by the user) and dense (i.e. one vector per pixel) velocity fields. The whole approach can be described as the minimization of one energy functional; there are no hidden model assumptions. Prior knowledge can be included into the energy functional in a natural way.

2.2.2. Particle Tracking Velocimetry

In contrast to CC-PIV methods, Particle Tracking Velocimetry (PTV) methods first extract the individual particle positions and then try to solve a *correspondence problem*: particles in one frame have to find their counterpart in a second frame.

[8] Depending on image quality and flow field gradients $\approx 0.01 - 0.1$ pixels

Individual-Particle Detection

In order to track individual particles, these particles first have to be extracted out of the gray-value structure of the image. Many authors have concentrated on this topic (a comparison of different particle detection approaches can be found in [OL00]).
There are two main types of particle detection approaches:

- **Region Growing Approaches [Maa92b]:** First, the whole image plane is scanned for local intensity maxima. Neighboring points of these germ points are iteratively added to the individual regions if they are above a certain threshold. Finally, the centroids of the individual regions are computed.

- **Particle Mask Correlation [ET99, SSKH00]:** In order to identify the central positions of the tracer particles, a Gaussian particle mask is used (typically of size 3×3 or 5×5). This mask is centered on all pixels in the image plane and cross-correlations are performed. Peaks in the correlation plane indicate particle centers. For sub-pixel accuracy, the same methods can be used as described in sec. 2.2.1 (iv).

Correspondence Problem

PTV methods are traditionally either based on nearest-neighbor search with geometrical constraints (using four or more consecutive frames) [KSS89, HC91], or on binary-image cross correlation (two frames) [UYO89], which computes the cross-correlation between regions around particles in the first and in the second frame. More recent approaches include relaxation methods that analyze the probability of particle matching [BL96, OL00], and genetic algorithms that evaluate different pairing schemes based on local morphology conservation or the constraint of vanishing divergence (for incompressible fluids) [SM98, DHS04].
Basically, all these methods have two assumptions in common:

- **Small displacements:** While nearest-neighbor search algorithms directly rely on small displacements from one frame of an image sequence to the next (in proportion to the particle density), binary-image correlation methods and relaxation methods both search for possibly corresponding particle images in a certain "tracking range".

- **Smoothness of motion:** Nearest-neighbor search algorithms assume that a particle changes its motion only smoothly during an image sequence. A similar assumption that tacitly underlies binary-image correlation methods is that the particles within a correlation window move with the same speed (if they do not, the correlation peak is less pronounced and the estimates become less reliable). Finally, using relaxation methods, a matching is considered probable if the movement of particles in a certain region can be reduced to a simple translation.

Note that PTV methods are in principle capable of yielding a higher resolution than PIV methods, as it is not necessary to average over regions in the image (i.e. interrogation windows). This requires, however, that the particle centroids are detected very accurately. On the other hand, PTV methods often fail if the overall motion is very large. Some algorithms therefore combine PIV and PTV (*super resolution analysis*, cf. [SR01]).
Furthermore, in 3D, PTV can be supported and combined with stereoscopic analysis and 3D reconstruction, leading to high-resolution 3D3C vector fields (e.g. [Maa92a, Maa92b]).

Conclusion

We have seen that PTV evaluation consists of two steps: detecting the individual particles and tracking them from one frame to the next.
Later in this thesis, we will concentrate on the *second* step: We introduce a variational method for PTV. We combine a discrete non-differentiable particle matching term with a continuous regularization term. The advantage of our approach is that physical constraints can be incorporated *directly*.

3. Standard Variational Methods for Motion Estimation

This chapter is composed as follows: First we introduce variational approaches in general and give two short examples as to how this type of method is applied in different scenarios (sec. 3.1). Then we go into variational approaches for motion estimation in particular. We present the pioneering work of Horn&Schunck and its numerical realization, and underline its advantages compared to cross-correlation methods.

3.1. Variational Methods in Computer Vision

Many relevant computer vision tasks can be conveniently solved by minimizing some energy measure. In this section we will give two short examples of how variational methods can be used to solve typical computer vision problems:

(i) Image smoothing,

(ii) Image segmentation.

A third problem that is typically solved with variational methods is motion estimation. It will be described in detail in sec. 3.4. For more details about variational methods for typical computer vision problems, we refer to [Sch99] and references therein.

(i) Image Smoothing Given is a (noisy) image $I(x_1, x_2)$. The task is to denoise the image, i.e. to smooth it. This can be easily performed by minimizing the energy functional

$$J(g) = \frac{1}{2}\int_\Omega \underbrace{(I-g)^2}_{\text{Data Term: Matching}} + \alpha \underbrace{|\nabla g|^2}_{\text{Regularization Term: Smoothness}} dx\ . \tag{3.1}$$

The function g that we search should on the one hand be close to the input data I and on the other hand it should be smooth. Note that α is a regularization parameter ($0 < \alpha \in \mathbb{R}$) that controls the degree of smoothness.
(3.1) can be considered as the simplest member of a whole class of computer vision problems that consist of two terms: a *data term* that measures the consistency with input data (usually images) and a *regularization term* that introduces prior knowledge, usually about smoothness or regularity.

The term *variational approach* bases on the fact that, in order to minimize (3.1), one requires the first variation of (3.1) to be zero:

$$\frac{\partial J(g+\epsilon\tilde{g})}{\partial \epsilon}\Big|_{\epsilon=0} = \int_\Omega (I-g)\tilde{g} + \alpha \nabla g^\top \nabla \tilde{g} dx = 0\ . \tag{3.2}$$

Figure 3.1.: **Left:** Noisy image I. **Middle:** Smoothed image g ($\alpha = 0.5$). **Right:** Smoothed image g ($\alpha = 5$).

Discretization of (3.2) yields a positive definite system that can be solved by some corresponding (iterative) solver. Appendix A gives details about the discretization process for elliptic systems.

Note that various extensions to the prototypical approach (3.1) have been suggested (e.g. nonlinear filtering, deconvolution, ...). For a review, we refer to [WS01a].

(ii) Image Segmentation Given is again a gray value image $I(x_1, x_2)$. A typical computer vision problem is to partition the image into multiple regions: Within each region, the individual gray values should be similar while gray value jumps should only occur at the curves that separate the individual partitions.

Let C denote the discontinuity set that serves as an interface (i.e. at the boundary of objects). Mumford and Shah [MS85, MS88] have proposed the following functional to obtain a segmented image g:

$$J(g, C) = \alpha \int_{\Omega} (I - g)^2 dx + \beta \int_{\Omega \setminus C} |\nabla g|^2 dx + \gamma |C| \,, \tag{3.3}$$

where $\alpha, \beta, \gamma > 0$. Note that the first term is minimal if the reconstructed image g is close to the image data I. The second term states that g should be smooth everywhere except for the interfaces that separate the individual partitions. Finally, the total edge length $|C|$ should be small. Note that minimizing this type of energy functional is intricate, and uniqueness of the minimizer is generally not given. A number of extensions to the prototypical approach (3.3) have been suggested (e.g. [BCG96, BG96, KLM94, CV01]). A discussion of these methods can be found, e.g., in [Par06]. Figure 3.2 shows the segmentation of the Andromeda Nebula as an example.[1]

[1] The author thanks Christian Gosch for providing the segmented images.

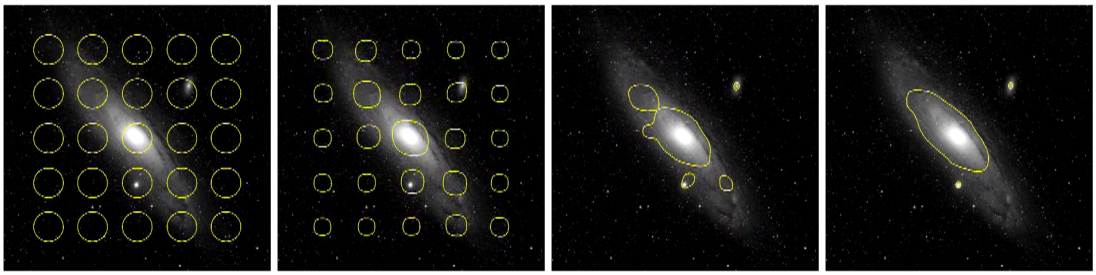

Figure 3.2.: Segmentation of the Andromeda Nebula. An artificial time t is introduced and an energy functional related to (3.3) is iteratively minimized. The images denote intermediary results at $t = 1$ (curve initialization), $t = 100$, $t = 200$, and the final evolution of the curves at $t = 300$.

3.2. Motion Estimation: The Optical Flow Constraint

Let $I(x_1, x_2, t)$ denote the gray value recorded at location $(x_1, x_2)^\top$ and time t in the image plane. Recall the basic assumption underlying most approaches to motion estimation – the conservation of I over time.

$$I(x_1 + u_1\Delta t, x_2 + u_2\Delta t, t + \Delta t) = I(x_1, x_2, t) \; . \tag{3.4}$$

We have already seen that a common approach to estimating the *optical flow vector* $(u_1, u_2)^\top$ at some fixed location $(\overline{x_1}, \overline{x_2})^\top$ on the image grid $(x_1, x_2)^\top = (k_1\Delta x_1, k_2\Delta x_2)^\top$, $k_1, k_2 \in \mathbb{Z}$, is to assume u_1 and u_2 to be constant within a local spatial area $N(\overline{x_1}, \overline{x_2})$ around $(\overline{x_1}, \overline{x_2})^\top$ and to minimize[2]

$$\sum_{k_1,k_2 \in N(\overline{x},\overline{y})} \big[I(k_1 + u_1\Delta t, k_2 + u_2\Delta t, t + \Delta t) - I(k_1, k_2, t)\big]^2 \tag{3.5}$$

as a function of u_1 and u_2. Assuming additionally that $\sum_{k_1,k_2 \in N(\overline{x_1},\overline{x_2})} I(k_1, k_2, t)^2$ does not vary with $(\overline{x_1}, \overline{x_2})^\top$, the minimizing values of u_1, u_2 maximize the correlation function

$$\sum_{k_1,k_2 \in N(\overline{x_1},\overline{x_2})} I(k_1 + u_1\Delta t, k_2 + u_2\Delta t, t + \Delta t)\; I(k_1, k_2, t) \; .$$

The first major difference to variational approaches is that the latter explicitly take into account smooth changes of the flow $(u_1, u_2)^\top$ at time t as a function of x_1 and x_2: $u_1 = u_1(x_1, x_2), u_2 = u_2(x_1, x_2)$. A continuously formulated expression analogous to (3.5) then reads:

$$\int_\Omega \big[I(x_1 + u_1(x_1, x_2)\Delta t, x_2 + u_2(x_1, x_2)\Delta t, t + \Delta t) - I(x_1, x_2, t)\big]^2 dx. \tag{3.6}$$

Note that since we do no longer assume u_1 and u_2 to be piecewise constant according to a subdivision of the visible section Ω of the image plane into interrogation areas, we integrate over the entire image domain Ω, observing the Neumann border conditions. From the viewpoint

[2] Without loss of generality we take $\Delta x_1 = \Delta x_2 = 1$.

of variational analysis and algorithm design, formulation (3.6) is less favorable because the dependency on u_1 and u_2 is highly non-convex. A common way around this difficulty is (i) to further simplify the objective function so as to obtain a mathematically tractable problem, and (ii) to apply the resulting variational approach to a multi-scale representation of the image data I (see section 4.1.3) so that the following approximation becomes valid:

$$\begin{aligned} & I(x_1+u_1\Delta t, x_2+u_2\Delta t, t+\Delta t) \\ & \quad \approx I(x_1,x_2,t) + \partial_{x_1} I(x_1,x_2,t) u_1 \Delta t + \partial_{x_2} I(x_1,x_2,t) u_2 \Delta t + \partial_t I(x_1,x_2,t)\Delta t \\ & \quad = I(x_1,x_2,t) + \nabla I(x_1,x_2,t) \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \Delta t + \partial_t I(x_1,x_2,t)\Delta t \ , \end{aligned} \tag{3.7}$$

where the spatial and temporal derivatives of I can be estimated locally using FIR filters. Inserting this approximation into (3.4) (and dropping the argument (x_1, x_2, t) for convenience) yields:

$$\nabla I \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_t I = 0 \ . \tag{3.8}$$

According to computer vision literature, this is the so-called *Brightness Change Constraint Equation (BCCE)* which, with a differential formulation, merely reflects our basic assumption (3.4) made in the beginning:

$$\frac{d}{dt} I(x_1,x_2,t) = 0 = \nabla I \cdot \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} + \partial_t I \ .$$

Using (3.7) and (3.8), the objective function (3.6) becomes:

$$\int_\Omega \left[\nabla I \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_t I \right]^2 dx \ . \tag{3.9}$$

Note that this objective function now depends *quadratically* on the two *functions* $u_1(x_1, x_2)$ and $u_2(x_1, x_2)$, which is much more convenient from the mathematical point-of-view. So far, the transition to a continuous setting has led us to the formulation (3.9) which has to be minimized with respect to arbitrary functions u_1 and u_2. Clearly, this problem is not well-posed as yet because *any* vector field with components $\nabla I \cdot (u_1, u_2)^\top = -\partial_t I, \forall x_1, x_2$, is a minimizer. This effect is called *aperture problem*: Motion that is perpendicular to the gradient of the energy function cannot be perceived. To realize this problem, consider the simple example of looking at a white piece of paper that is moving in front of a black background (cf. fig. 3.3). Now consider observing just the vertical transition between white and black through a porthole: All motions with equivalent vertical components will appear identical to us.

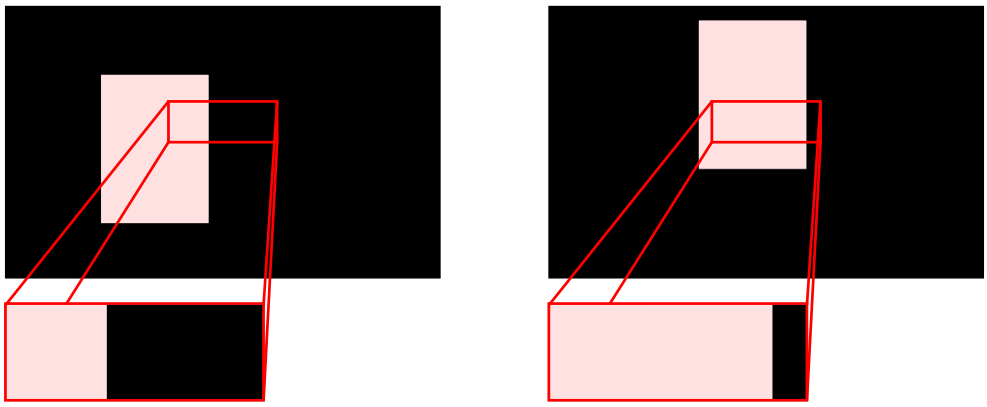

Figure 3.3.: Aperture Problem: White piece of paper moving in front of a black background. When considering only the framed part of the image, all motions with equivalent vertical velocity components will appear identical.

3.3. Local Approaches for Optical Flow Estimation

In the preceding section we have seen in the preceding section that the aperture problem prohibits a direct solution of (3.9) and that we therefore need additional constraints. The simplest method to overcome this problem is to assume that the velocity field is constant within a certain neighborhood $\mathcal{N}$ (i.e., $u(x) = u(x_0), \forall x \in \mathcal{N}(x_0)$). We can therefore minimize

$$J(u) = \sum_{x \in \mathcal{N}(x_0)} G_\rho(x - x_0)(\nabla I(x) \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_t I(x))^2 \,, \tag{3.10}$$

where G_ρ is a Gaussian distribution with variance ρ. It is clear that the minimum of this energy is at

$$G_\rho * \begin{pmatrix} (\partial_{x_1} I)^2 & \partial_{x_1} I \, \partial_{x_2} I \\ \partial_{x_1} I \, \partial_{x_2} I & (\partial_{x_2} I)^2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = -G_\rho * \begin{pmatrix} \partial_t I \, \partial_{x_1} I \\ \partial_t I \, \partial_{x_2} I \end{pmatrix} \,. \tag{3.11}$$

Note that (3.11) is not necessarily uniquely determinable. In areas where the gray value is, e.g., homogeneous or at image edges there will not be a unique solution. Furthermore, as already seen in sec. 2.2.1, the assumption of a constant velocity inside some neighborhood is often violated.

Note that there is vast research on *local* optical flow methods, which is, however, beyond the scope of this thesis. For a review on different types of local optical flow approaches and extensions, we refer to [Jäh97, HGSJ99].

Recently, enhanced brightness change models have been introduced, that exchange the brightness change constraint equation (3.8) by terms that model (physics-based) brightness variation (cf. [HF01] and references therein). These methods have been successfully applied to fluid imagery [SDG02]. The authors show that even 3D information can be extracted out of 2D image data, provided that the depth information can be related to the brightness model (*depth from diffusion*).

3.4. The Method of Horn&Schunck

Rather than to consider vector fields which are piecewise constant within interrogation areas, we want to follow the ideas presented in section 3.1: We rule out too irregular vector fields by additionally minimizing the magnitudes of the spatial gradients of u_1 and u_2:

$$J(u_1, u_2) = \int_\Omega \left\{ \left[\nabla I \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_t I\right]^2 + \lambda\left(|\nabla u_1|^2 + |\nabla u_2|^2\right) \right\} dx \;, \quad 0 < \lambda \in \mathbb{R} \;. \tag{3.12}$$

Parameter λ is either a user-parameter or can be determined as a Lagrange multiplier related to either of the constraints

$$\int_\Omega \left[\nabla I \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_t I\right]^2 dx = \alpha \;,$$
$$\int_\Omega \left(|\nabla u_1|^2 + |\nabla u_2|^2\right) dx = \beta \;,$$

provided either of the variables α or β is known. The discussion of this interpretation of the regularization parameter is, however, beyond the scope of this manuscript, and we regard λ as a user-parameter. A large value for λ leads to a very smooth flow field, whereas the smoothness decreases for smaller values for λ. At locations with $|\nabla I| \approx 0$ (i.e. untextured regions), no reliable flow can be estimated from the data term. At these locations, the smoothness term solves this problem by filling in information from the neighborhood, leading to a dense flow field.

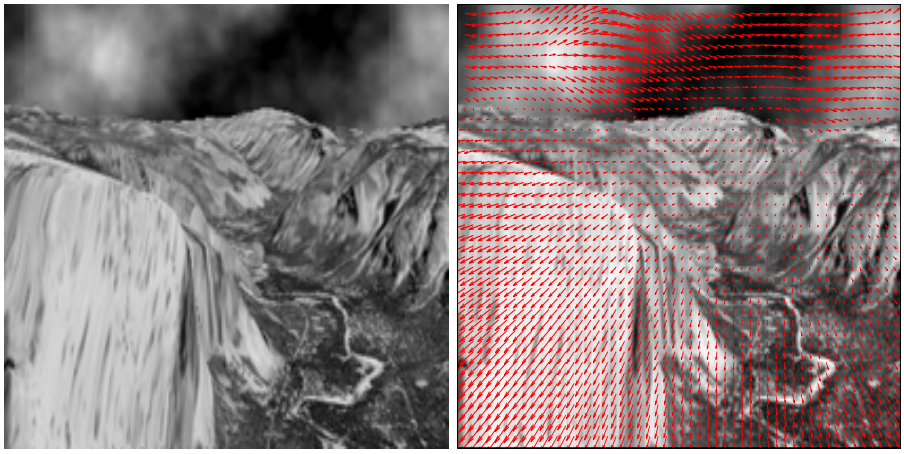

Figure 3.4.: Flight through the Yosemite Valley. A frame from a typical synthetic image sequence that is often used to compare different optical flow based techniques. In the right image, you see the solution of the simple Horn&Schunck approach as presented in sec. 3.4. The flight through the valley induces a divergent velocity field, while the clouds move to the right. Note that the rock in the left part of the image ("El Capitan") is closer to the camera – that is why its optical flow vectors are longer.

Figure 3.4 shows a typical example for the type of image sequences, where variational optical

flow approaches are usually applied. The Horn&Schunck approach can reconstruct the target velocity field reasonably well.

3.5. Optimization and Discretization

Under mild conditions with respect to the image sequence data I it can be shown [Sch91] that the functional (3.12) is strictly convex. We will use the *Finite Element Method (FEM)* to discretize (3.12). For details we refer to [Cia78, Sch99] and appendix A. The unique globally minimizing vector field $\big(u_1(x_1,x_2), u_2(x_1,x_2)\big)^\top$ is determined by the variational equation

$$a\big((u_1,u_2)^\top,(\tilde{u}_1,\tilde{u}_2)^\top\big) = b\big((\tilde{u}_1,\tilde{u}_2)^\top\big)\ , \quad \forall \tilde{u}_1,\tilde{u}_2\ , \tag{3.13}$$

where

$$a\big((u_1,u_2)^\top,(\tilde{u}_1,\tilde{u}_2)^\top\big) = \int_\Omega \Big\{ \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \cdot \nabla I \nabla I \cdot \begin{pmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{pmatrix} + \lambda\Big(\nabla u_1 \cdot \nabla \tilde{u}_1 + \nabla u_2 \cdot \nabla \tilde{u}_2\Big)\Big\} dx\ , \tag{3.14}$$

$$b\big((\tilde{u}_1,\tilde{u}_2)^\top\big) = -\int_\Omega \partial_t I \nabla I \cdot \begin{pmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{pmatrix} dx\ . \tag{3.15}$$

The simplest discretization is obtained by choosing a regular triangulation of the image domain Ω and attaching to each pixel position a piecewise linear basis function $\phi(x_1,x_2)$ for each function $u_1, u_2, \tilde{u}_1, \tilde{u}_2$, as illustrated in figure 3.5. Indexing each pixel position (k,l) by $1,2,\ldots,N$

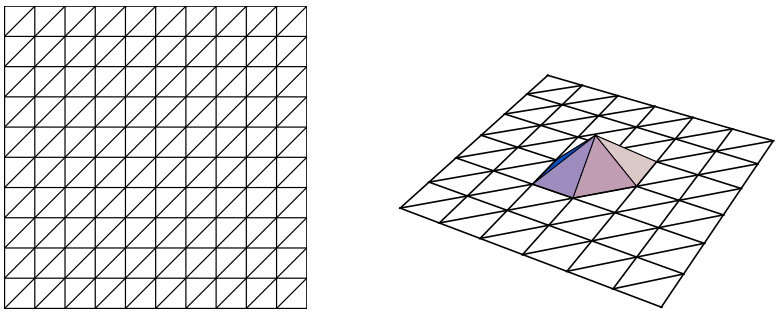

Figure 3.5.: **Left:** Uniform triangulation of the image domain Ω. **Right:** Basis function $\phi_i(x_1,x_2)$ attached to pixel position i.

we thus have

$$u_1(x_1,x_2) = \sum_{i=1}^{N} u_1^i \phi^i(x_1,x_2)\ ,$$

and similarly for $u_2, \tilde{u}_1, \tilde{u}_2$. Hence, each of the functions $u_1, u_2, \tilde{u}_1, \tilde{u}_2$ is represented by N real variables. To simplify notation, we use the same symbols to denote these vectors: $u_1, u_2, \tilde{u}_1, \tilde{u}_2 \in \mathbb{R}^N$. Inserting into (3.13) leads to:

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \cdot A \begin{pmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{pmatrix} = b \cdot \begin{pmatrix} \tilde{u}_1 \\ \tilde{u}_2 \end{pmatrix}\ , \quad \forall \tilde{u}_1,\tilde{u}_2\ ,$$

hence:

$$A \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = b \; . \tag{3.16}$$

The $2N \times 2N$–Matrix A factorizes into

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{12}^\top & A_{22} \end{pmatrix} \; ,$$

where by virtue of (3.14):

$$\begin{aligned} (A_{11})_{k,l} &= a\big((\phi_k, 0)^\top, (\phi_l, 0)^\top\big) \\ (A_{12})_{k,l} &= a\big((\phi_k, 0)^\top, (0, \phi_l)^\top\big) \\ (A_{22})_{k,l} &= a\big((0, \phi_k)^\top, (0, \phi_l)^\top\big) \; . \end{aligned}$$

Analogously, the $2N$–vector b factorizes into $b = (b_1^\top, b_2^\top)^\top$, where by virtue of (3.15):

$$\begin{aligned} (b_1)_k &= b\big((\phi_k, 0)^\top\big) \\ (b_2)_k &= b\big((0, \phi_k)^\top\big) \; . \end{aligned}$$

The linear system (3.16) is sparse and positive definite. Thus u_1, u_2 can be conveniently computed by some corresponding iterative solver [Hac93]. For numerical details, we refer to appendix A.

3.6. Discussion

In view of the limitations of CC-PIV mentioned in section 2.2.1, we point out the following features of the variational approach (3.12):

+ The approach is formulated in terms of *functions* u_1 and u_2, and hence, by definition, provides motion estimates $\big(u_1(x_1, x_2), u_2(x_1, x_2)\big)^\top$ at *any* point $(x_1, x_2)^\top \in \Omega \subset \mathbb{R}^2$.

+ Spatial variation of u_1, u_2 is merely constrained by a *global* penalty term (i.e. the second term in (3.12)). Accordingly, the motion field $(u_1, u_2)^\top$ may exhibit spatial variations of different strengths depending on the evidence provided by the spatio-temporal image sequence data I.

+ The approach is intrinsically non-local and allows to incorporate spatial context in a mathematically convenient way by means of functionals depending on u_1, u_2 and corresponding derivatives.

Note, however, that there are some potential drawbacks of the presented prototypical variational optical flow approach that we will have to analyze in following chapters:

- The image structure of typical PIV images is very special. It is not clear if optical flow approaches are capable of yielding reliable velocity fields.

- The approach as formulated in (3.12) will only yield reliable velocity estimates if the apparent motion between the two frames is smaller than 1 pixel. This is mainly due (i) to the limitations of the FIR filter that we use to compute the spatial and temporal gradients and (ii) to the fact that we truncated the Taylor series in (3.7).

- The fact that minimizing (3.12) yields one velocity vector per pixel does not necessarily mean that these estimates contain additional information about the high frequency components of a velocity field. This is questionable because the regularization term particularly penalizes the high frequency components of the velocity field.

While we show in chapter 4 that the first two problems can be easily overcome by coarse-to-fine strategies and iterated registration (cf. sec. 4.1.3), we will enhance the regularization term in chapter 5 and thoroughly investigate the interesting question of the highest achievable resolution in sec. 5.3.3.

4. Variational Fluid Motion Estimation: Data Term

In the preceding chapter, we have given an overview of variational methods for motion estimation. We have seen that these methods typically consist of a data term that measures the consistency between data and model, and a regularization term that introduces prior knowledge into the estimation and makes the system well-posed. In principle, the smoothness-term enforces coherent vector field structures, making corresponding post-processing steps in connection with local PIV-approaches obsolete.

We want to apply this class of approaches to PIV image pairs and sequences.
The emphasis of this chapter is on the data term, i.e. we will introduce data terms that are well-suited for the purpose of PIV/PTV. We will introduce two different types of data terms:

- In sec. 4.1 we introduce a data term for particle image velocimetry, i.e. a data term that operates on gray value images. The starting point of our research is the optical flow constraint (cf. sec. 3.2). We will adapt it for the purpose of PIV and evaluate it using synthetic and real PIV image pairs.

- In sec. 4.2 we introduce a novel variational approach for evaluating PTV image pairs and sequences in two and three dimensions (i.e. we track individual tracers over time). We replace the continuous data term of sec. 4.1 with a discrete non-differentiable particle matching term. The experimental evaluation shows that our method competes with three alternative approaches.

In both cases, we deliberately use the very simple first-order term of Horn&Schunck for regularization. This is done for simplicity and comparability. In chapter 5 we will present regularizers that are better suited for the typical velocity distributions present in PIV imagery and that are physically motivated.

4.1. Methods that Operate on Gray-Value Images

The objective of this section is to adapt a prototypical variational approach for motion estimation (i.e. the one presented in chapter 3) to the purposes of PIV. Note that we will only adapt the data term; special regularizers that use prior knowledge from fluid mechanics will be introduced in chapter 5.

In sec. 4.1.1–4.1.3, we will introduce enhancements to the variational framework presented in chapter 3.

Numerical experiments for benchmark image pairs and a comparison with alternative approaches especially designed for PIV-sequence evaluation will be presented in section 4.1.4.

The basic approach of Horn&Schunck has already been described in detail in sec. 3.4. In the current section, we want to focus on the adaption of Horn&Schunck's approach to the quite

specific gray value structure of PIV images. We will focus on three main aspects that affect the data term:

- **Invalid assumption of gray value conservation:** The optical flow constraint bases on the assumption of gray value conservation over time. This assumption is violated in case of changing gray values due to illumination changes. We will adapt the optical flow constraint in section 4.1.1.
- **Image pair analysis:** PIV imagery often consists of image pairs only, while standard variational approaches for motion estimation operate on whole image sequences. In sec. 4.1.2, we will reformulate the optical flow constraint and make it symmetric. We will show that this procedure improves the accuracy of optical flow estimation for image pairs.
- **Large displacements:** Algorithms for PIV evaluation should be able to resolve a large motion range. To handle large displacements, we will present a coarse-to-fine scheme in sec. 4.1.3.

In section 4.1.4, we will finally present numerical experiments for benchmark image pairs and a comparison with alternative approaches. We conclude in section 4.1.5 by indicating extensions of the presented approach within the variational framework.

4.1.1. Going Beyond the Assumption of Gray Value Conservation

There are a number of reasons why the gray value at a certain location will change from one PIV frame to the next:

- A particle that was located at a point x to a time t has traveled to position $x + u$ at time $t + \Delta t$.
- Particles have an out-of-plane velocity component. In case they travel out of the illuminated laser light sheet, their brightness fades; if they travel towards the illuminated plane, they gain in brightness.
- Problems with illumination: Often, the plane is not uniformly illuminated (e.g. due to experimental setup, dirt or properties of the expanded laser beam). Besides, the intensity of the laser beam tends to fluctuate over time.
- Properties of the camera: e.g. noise or quantization errors.

The standard optical flow constraint (3.9) deals with the illumination change introduced by the transport process (item 1). However, brightness changes due to 3D motion, illumination and image capturing are not modeled.

Extended Optical Flow Constraint

Let ρ be the density of some fluid and u its velocity. The physical equation of mass conservation is given with (cf. also sec. C.1)

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) = 0 \; .$$

If we assume that

- the image brightness I is proportional to the fluid density ρ and
- the 2D projection of the continuity equation holds true

we get

$$\frac{dI}{dt} + I \, \nabla \cdot u = 0 \, , \tag{4.1}$$

where $u = (u_1, u_2)^\top$. Note that in the case of a vanishing divergence of u, (4.1) is exactly the optical flow constraint. The extended optical flow constraint has been introduced in [Sch84] and applied to meteorological and fluid imagery in [WALL97, BHY00].

Integrated Continuity Equation

Corpetti et al. go into the fact that the continuity equation yields a velocity and not a displacement estimate [CMP02, CHA+05]. They assume that the velocity is constant between the two frames and integrate the continuity equation along the trajectories. This yields the minimization of

$$E(d) = \int_\Omega f(I(x + d(x), t + \Delta t) \exp(\nabla \cdot d(x)) - I(x, t)) dx \, , \tag{4.2}$$

where d is the displacement and f is some penalty function (e.g. L_2 norm). (4.2) is linearized and incorporated into some multi-resolution scheme (cf. sec. 4.1.3).

Note that while the extended optical flow constraint is physically motivated, it is not clear if the underlying assumptions are valid in PIV imagery. We will go into this question in sec. 4.1.4.

Modeling Illumination Changes

Recall the basic assumption behind optical flow:

$$I(x_1 + u_1 \Delta t, x_2 + u_2 \Delta t, t + \Delta t) = I(x_1, x_2, t) \, .$$

Let us exchange this term, that assumes that changements of gray value at a certain position in the image are only due to movement of objects in the image plane by

$$I(x_1 + u_1 \Delta t, x_2 + u_2 \Delta t, t + \Delta t) = I(x_1, x_2, t) + b(x_1, x_2, t) \, ,$$

where $b(x_1, x_2, t)$ is a scalar field that takes into account the above mentioned illumination changes. Note that the observed illumination changes arise from a multitude of effects (cf. above). We have chosen this very simple (additive) term for modeling illumination/ brightness changes, as the exact interaction of the different effects is usually not known and would require the incorporation of many new parameters.

The revised optical flow constraint therefore reads[1]

$$\nabla I \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_t I = b \, .$$

[1] We take $\Delta t = 1$ without loss of generality.

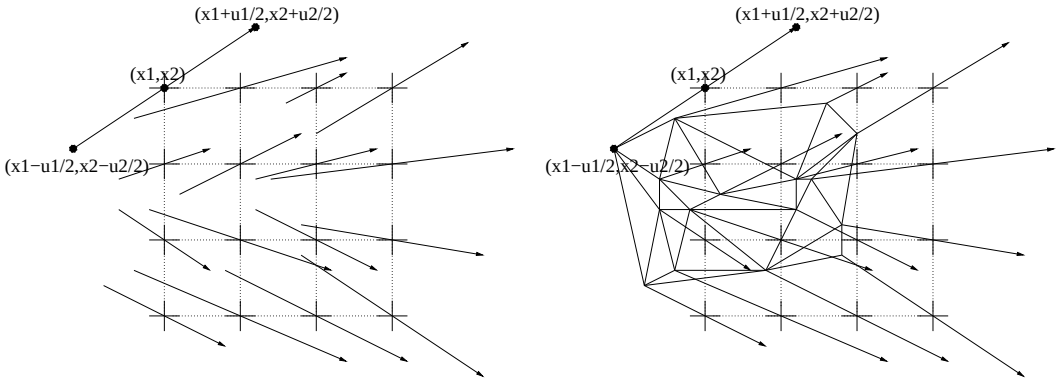

Figure 4.1.: Sample displacement field. Note that every displacement vector of position (x_1, x_2) starts at $(x_1 - u_1/2, x_2 - u_2/2)$ and ends at $(x_1 + u_1/2, x_2 + u_2/2)$. We perform Delaunay triangulation and interpolate the data back to the given grid.

If we expect $b(x_1, x_2)$ to vary smoothly, we can penalize strong variations of $b(x_1, x_2)$ in the smoothness term of a variational approach:

$$J(u_1, u_2, b) = \int_\Omega \left\{ \left[\nabla I \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \partial_t I - b\right]^2 + \lambda\left(|\nabla u_1|^2 + |\nabla u_2|^2\right) + \mu |\nabla b|^2 \right\} dx \ , \quad 0 < \lambda, \mu \in \mathbb{R} \ . \tag{4.3}$$

(4.3) can be perceived as a simplified version of the approach for robust motion estimation under varying illumination presented in [KMK05].

4.1.2. Symmetric Optical Flow Constraint

Traditional motion estimation tasks (e.g. for robot vision, driver's assistance systems or movie encoding) demand the analysis of whole image sequences. Temporal coherency facilitates the vector field recovery. Temporal derivatives can be estimated by using large temporal filters that suppress noise.
Due to camera restrictions, PIV imagery usually consists of image pairs only (cf. sec. 2.1.1). For this purpose, let us slightly change the optical flow constraint introduced in sec. 3.2: Given two images (i.e. samples (x_1, x_2) at times $t - \frac{1}{2}$, $t + \frac{1}{2}$ of a continuous 2D+time gray value function $I(x_1, x_2, t)$). We want to compute the displacement $(u_1(x_1, x_2), u_2(x_1, x_2))^\top$ [2] that maps both images onto each other [3]:

$$I(x_1 - \frac{1}{2}u_1, x_2 - \frac{1}{2}u_2, t - \frac{1}{2}) = I(x_1 + \frac{1}{2}u_1, x_2 + \frac{1}{2}u_2, t + \frac{1}{2}) \ . \tag{4.4}$$

Figure 4.1 shows that, by solving (4.4), we find displacement vectors $(u_1, u_2)^\top$ at positions (x_1, x_2) that map points that were at position $(x_1 - u_1/2, x_2 - u_2/2)$ in image 1 to points at position $(x_1 + u_1/2, x_2 + u_2/2)$ in image 2. Taylor series linearizations of both sides of (4.4)

[2] We assume that u_1 and u_2 are displacements that are constant in the interval $[t - \frac{1}{2}, t + \frac{1}{2}]$.
[3] Including the additional brightness function from the preceding section is straight-forward. For perceivability, we have not included the brightness function in this derivation.

yield

$$I(x_1, x_2, t - \frac{1}{2}) - \frac{1}{2}u_1 \frac{\partial I(x_1, x_2, t - \frac{1}{2})}{\partial x_1} - \frac{1}{2}u_2 \frac{\partial I(x_1, x_2, t - \frac{1}{2})}{\partial x_2}$$
$$= I(x_1, x_2, t + \frac{1}{2}) + \frac{1}{2}u_1 \frac{\partial I(x_1, x_2, t + \frac{1}{2})}{\partial x_1} + \frac{1}{2}u_2 \frac{\partial I(x_1, x_2, t + \frac{1}{2})}{\partial x_2} .$$

This yields the well-known brightness constancy equation

$$0 = I(x_1, x_2, t + \frac{1}{2}) - I(x_1, x_2, t - \frac{1}{2}) + \frac{1}{2}u_1 \Big(\frac{\partial I(x_1, x_2, t - \frac{1}{2})}{\partial x_1} + \frac{\partial I(x_1, x_2, t + \frac{1}{2})}{\partial x_1} \Big)$$
$$+ \frac{1}{2}u_2 \Big(\frac{\partial I(x_1, x_2, t - \frac{1}{2})}{\partial x_2} + \frac{\partial I(x_1, x_2, t + \frac{1}{2})}{\partial x_2} \Big) .$$

Note that the symmetric version of the brightness constancy constraint equation allows the use of *both* spatial derivatives of image 1 and image 2. This makes the algorithm more robust.
Due to the symmetric nature of the approach, changing the order of the two images solely changes the sign of the extracted vector field: The resulting vector field does not give the displacements at positions (x_1, x_2), but at positions $(x_1 - \frac{1}{2}u_1, x_2 - \frac{1}{2}u_2)$. In highly non-rigid scenarios, this fact is not negligible. If the resulting velocity field should be given at a regular grid, we will have to warp the velocity field and interpolate. This is done by

- building a grid using Delaunay triangulation with the warped pixel positions as vertices (cf. fig. 4.1), and
- interpolating back on the old regular grid using a simple linear interpolation scheme.

4.1.3. Coarse-to-Fine Motion Estimation

The accuracy of motion estimation critically depends on the magnitude of image motion. In fact, depending on the spatial image frequency, very large motions may even cause aliasing along the time frequency axis. For illustration, figure 4.2 shows a 1D-signal moving to the right at constant speed u:

$$I(x, t) = \sin\big(\omega_x(x - ut)\big).$$

Due to the Nyquist-condition $|\omega_t| < \pi$ (with $\omega_t := \omega_x u$), only motions up to $|u| < \pi/\omega_x$ are correctly represented by samples of the signal.[4] Faster motions lead to aliasing. In other words, for a fixed global velocity, spatial frequencies moving more than half of their period per frame cause temporal aliasing. In practice, this upper bound has to be lowered because derivatives of the signal can only be robustly estimated in connection with low-pass filtering.

As a remedy, we first compute a coarse motion field by using only low spatial frequency components and "undo" the motion, thus roughly stabilizing the position of the image over time. Then the higher frequency sub-bands are used to estimate optical flow on the warped sequence. Combining this "optical flow correction" with the previously computed optical flow yields a refined overall optical flow estimate. This process may be repeated at finer and finer spatial scales until the original image resolution is reached [KMW96, Sim93]. A standard technique for generating multi-scale representations in this context is to construct an image

[4] Without loss of generality we assume sampling rates $\Delta x = \Delta t = 1$.

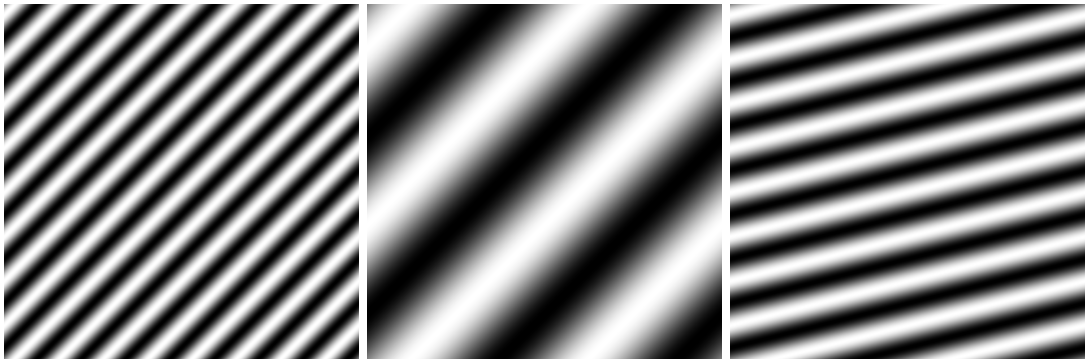

Figure 4.2.: Moving signal $g(x,t) = \sin(\omega_x(x - ut))$ as gray value image with different parameters ω_x (spatial frequency) and u (velocity). **Left:** $\omega_x = \pi/4$, $u = 1$, **Middle:** $\omega_x = \pi/16$, $u = 1$, **Right:** $\omega_x = \pi/16$, $u = 4$. Temporal frequency is affected by both spatial frequency ω_w and velocity u.

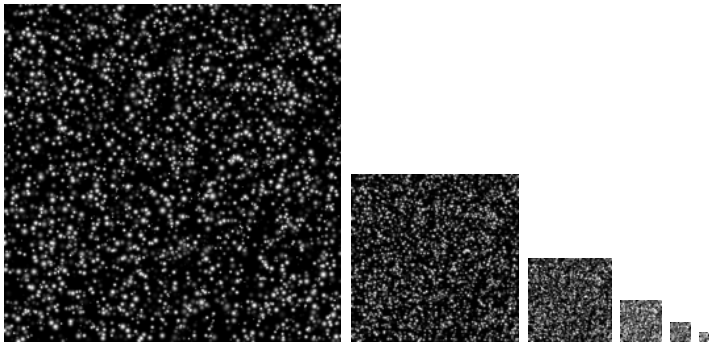

Figure 4.3.: Image Pyramid: Each level in the pyramid is a sub-sampled version of the level below convolved with a Gaussian filter.

pyramid (figure 4.3) by recursively applying low-pass filtering and sub-sampling operations. Note that the images at different scales are represented by different sampling rates. Thus, the same derivative filters may be used at each scale and we do not have to design multiple derivative filters, one for each different scale. Let us define the pyramid representation of a generic image I of size $n_{x_1} \times n_{x_2}$. Let $I^0 = I$ be the "$zero^{th}$" level image. This image is essentially the highest resolution image (the raw image). The image width and height at that level are defined as $n^0_{x_1} = n_{x_1}$ and $n^0_{x_2} = n_{x_2}$. The pyramid representation is then built in a recursive fashion: Compute I^1 from I^0, then compute I^2 from I^1, and so on ... . Let $k = 0, 1, 2, ..., L-1$ be a generic pyramidal level, and let I^k be the image at level k. $n^k_{x_1}$ and $n^k_{x_2}$ denote the width and the height of I^k. First the low-pass filter $[1/4\ 1/2\ 1/4] \times [1/4\ 1/2\ 1/4]^\top$ is used for image anti-aliasing before image sub-sampling. Then a bilinear interpolation performs the adaptation to the new coarser grid, as every new vertex is located exactly in the middle of four finer vertices (if the respective image size is even-numbered, cf. figure 4.4). This procedure results in a convolution mask of $[1/8\ 3/8\ 3/8\ 1/8] \times [1/8\ 3/8\ 3/8\ 1/8]^\top$. In the first step the optical flow between the top level images I_1^{L-1} and I_2^{L-1} (lowest frequency images) is computed, using the variational

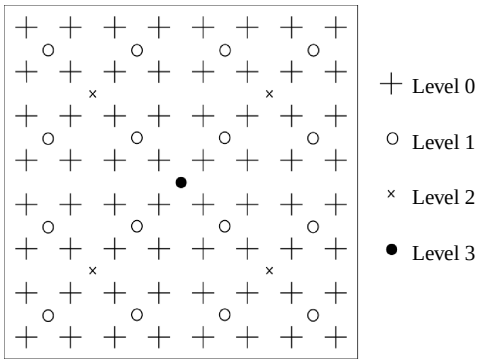

Figure 4.4.: Image Pyramid: Location of the vertices in the respective levels.

approach of equation (4.3). The computed coarse-level flow field must then be projected onto the next finer pyramid level ($L-2$). This flow field estimate is used to warp both images onto each other. The warped versions (I^*) therefore read

$$I^*(x_1, x_2, t^-) := I^{L-2}(x_1 - \frac{1}{2}u_1, x_2 - \frac{1}{2}u_2, t - \frac{1}{2})$$
$$I^*(x_1, x_2, t^+) := I^{L-2}(x_1 + \frac{1}{2}u_1, x_2 + \frac{1}{2}u_2, t + \frac{1}{2}) \; .$$

Note that, as $\big((x_1 \pm \frac{1}{2}u_1), (x_2 \pm \frac{1}{2}u_2)\big)$ usually does not lie on our regular grid, we have to interpolate. This is done using second order spline interpolation. Next, we compute a new and finer flow field between the images $I^*(x_1, x_2, t^-)$ and $I^*(x_1, x_2, t^+)$. While the expression to be minimized is analogous to (4.3), we now have to distinguish between the overall velocity ($(u_1(x_1, x_2), u_2(x_1, x_2))^\top$, which should be smooth) and the velocity update ($(u_1^{up}(x_1, x_2)$, $u_2^{up}(x_1, x_2))^\top$, that is to be measured):

$$J(u_1, u_2, b) = \int_{\Omega \setminus \Omega_0} \Big[\nabla I \cdot \begin{pmatrix} u_1^{up} \\ u_2^{up} \end{pmatrix} + \partial_t I - b^{up}\Big]^2 dx + \lambda \int_\Omega \big(|\nabla u_1|^2 + |\nabla u_2)|^2\big) + \mu |\nabla b|^2 dx \; ,$$
$$0 < \lambda, \mu \in \mathbb{R} \; . \tag{4.5}$$

Note that Ω_0 denotes those boundary regions in the image, where no image derivatives can be computed due to motion over the image boundaries.
Substituting $(u_1^{up}, u_2^{up})^\top$ with $(u_1 - u_1^{old}, u_2 - u_2^{old})^\top$ (and b^{up} with $b - b^{old}$), where (u_1^{old}, u_2^{old}) is just the (projected) result of the preceding estimation step on level $L-1$ (and resp. for b^{old}), the unique flow field minimizing (4.5) is the refined estimate of the overall flow field. This process is repeated for each level of the pyramid until the finest pyramid level k^0 has been reached. In the experimental evaluation section below, we will refer to this approach as Horn&Schunck Multi-Resolution (**H&S R**). So far, we have introduced a *dyadic* pyramid structure which is equivalent to using low-pass filters with bandwidths $\frac{\Omega}{2^{L-1}}, \frac{\Omega}{2^{L-2}}, ..., \frac{\Omega}{2^1}, \frac{\Omega}{2^0}$ combined with sub-sampling. Now we introduce additional filters that slice the bandwidth into even smaller pieces, e.g. $\Omega/4$, $3/8\Omega$, $\Omega/2$, $3/4\Omega$, Ω. In order to implement these extra steps which do not fit into the dyadic pyramid structure, we apply at each pyramid level pre-filters when estimating derivatives. Figure 4.5 shows the effect on a typical particle image: the lower the cut-off frequency of the pre-filter,

the more the particles seem to melt down and form a smooth gray value structure. A coarse motion estimate can reliably be computed using this structure. Then, we update and refine the motion field in the same way as described in detail for the multi-resolution case (i.e. iterated registration) using the less low-pass filtered image derivatives. Figure 4.6 shows the frequency

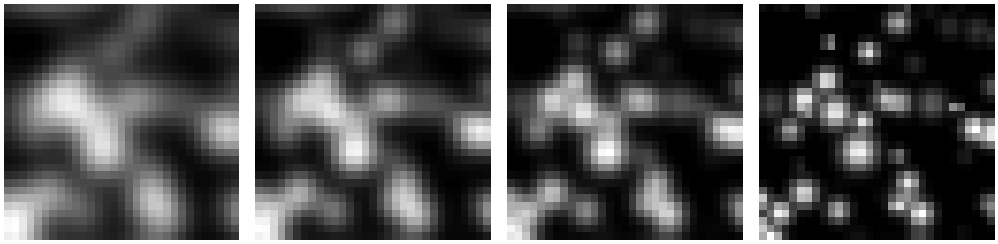

Figure 4.5.: A sample particle image in different scale levels.

spectra of the Gaussian filters we apply, for the case of five scale-space levels. In practice, we use nine scale-space levels and thus nine different filters with cut-off frequencies of $\frac{\pi}{2}$, $\frac{9}{16}\pi$, $\frac{5}{8}\pi$,$\frac{11}{16}\pi$, $\frac{3}{4}\pi$, $\frac{13}{16}\pi$, $\frac{7}{8}\pi$, $\frac{15}{16}\pi$, π. An inverse Fourier Transform yields the filter coefficients. Low pass filtering with cut-off frequencies below $\pi/2$ is not necessary, since this is what the anti-aliasing filter of the preceding lower resolution level has already done.
Below, we will refer to this combined approach as Horn&Schunck Multi-Resolution + Multi-Scale (**H&S R+S**).

4.1.4. Experimental Evaluation

There are two main concerns for this section: First, we want to validate our approach and show that our suggested improvements to the simple Horn&Schunck approach (especially symmetric warping and the additional brightness correction term) do actually improve the accuracy. Second, we report comparisons of the variational approach with three other approaches for various data sets.
Before discussing the results, we first describe the data sets used for the comparison, the alternative approaches (besides the variational approach) and corresponding parameter setting and quantitative error measures.

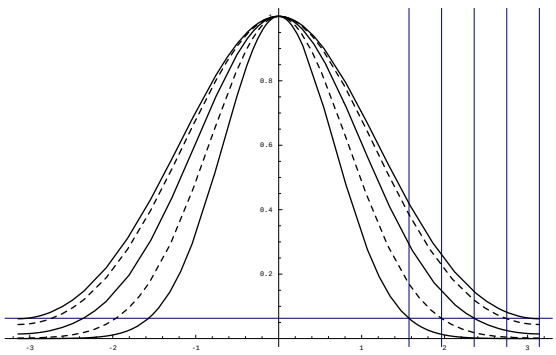

Figure 4.6.: Gaussian filters with cut-off frequencies of $\pi/2$, $5/8\pi$, $3/4\pi$, $7/8\pi$, π.

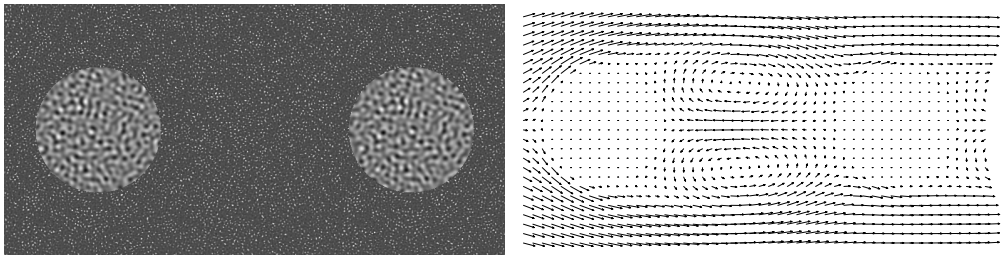

Figure 4.7.: Quénot image pair. **Left:** Synthetic particle image. **Right:** Exact velocity field (av. 7.58 px/frame)

Data

The experimental evaluation was carried out on the basis of the following data sets:

- **Simple Rotation:** The target vector field obeys the simple analytical function $u_1(x_1, x_2) = x_2$, $u_2(x_1, x_2) = -x_1$. To generate the first set of synthetic images (case A, fig. 4.8), we used the same methods as described in [ONK00a]. Note, however, that there is no out-of-plane velocity in this example. The image size is 256×256, the velocity field is scaled to have a maximum displacement of 14.14 pixels (at the four corners). For the second pair of images (case B, fig. 4.10), we mimicked local changes in illumination, by adding a Gaussian-shaped patch to the brightness function of the second image.

- **Quénot image pair:** This set of artificial benchmark image pairs was introduced in [QP98] and is available on the Internet. The analyzed velocity field (av. velocity = 7.58 pixels/frame) is taken from a numerical solution, obtained for two-dimensional flow around a pair of cylinders (figure 4.1.3).
  We examined ten different test cases being part of the following four classes:

  - **Perfect:** "Perfect" case means that the second image was computer-generated from the first image and the target flow field.
  - **Noise N%:** Additionally to the "perfect" case, noise was superimposed for all images.
  - **Add/Rm N%:** The specified percentage of particles was randomly removed and the same amount of particles was randomly added.
  - **Mixed N%:** In this case all images were corrupted by both types of errors (Noise N% and Add/Rm N%) simultaneously.

- **VSJ:** In 1999, the Visualization Society of Japan (VSJ) published standard PIV images on their website [ONK00a]. There are eight different computer-generated standard image pairs. They differ from each other in image features as well as in flow field attributes.

  Table 4.1 lists the parameters of these standard images: the number of particles that are present in the images, the particle diameter and the standard deviation of the particle diameter, the average image velocity and the out-of-plane velocity. The average image velocity defines the particle displacement between two successive images. The target

velocity field (figure 4.11) is scaled in order to achieve a certain average image velocity. This is equivalent to adapting the temporal sampling rate.

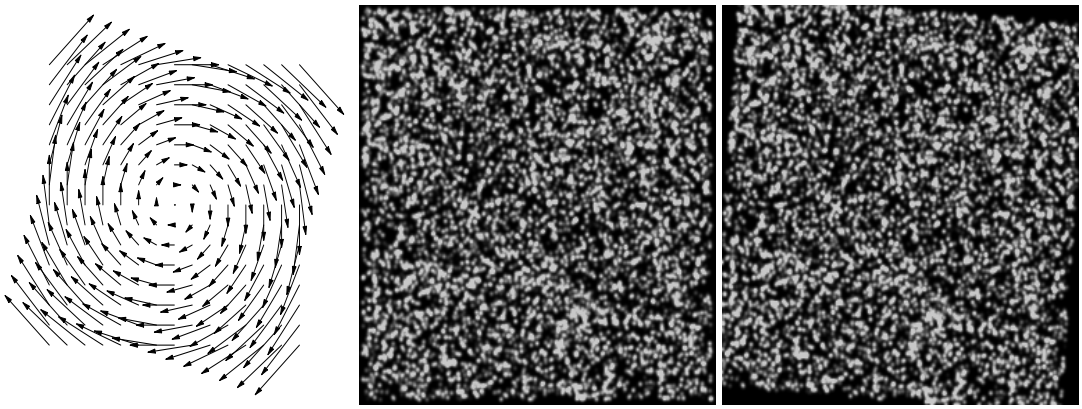

Figure 4.8.: Simple Rotation (case A). Size: 257×257. Maximum displacement (at the four corners): 14.14 px

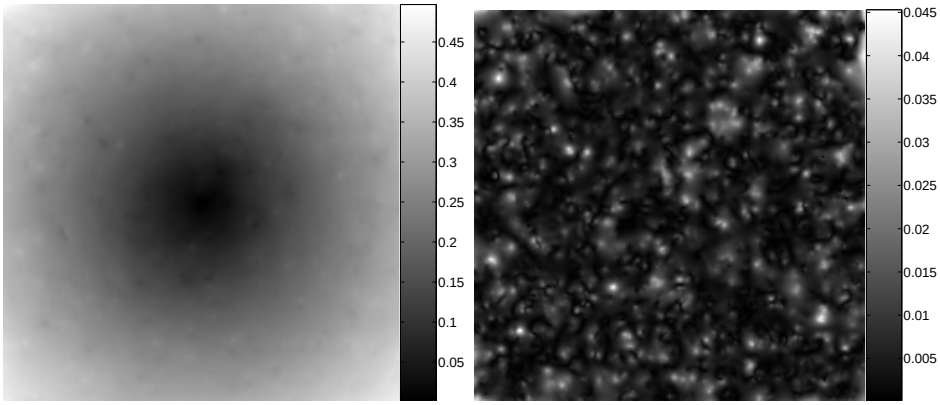

Figure 4.9.: **Left:** Displacement error when assuming $u_1(x_1 - \frac{1}{2}u_1, x_2 - \frac{1}{2}u_2) = u_1(x_1, x_2)$ (and resp. for u_2). Mean error: 0.2795 pixels. Note that the errors are large at positions with large motion. **Right:** Displacement error after additional vector field warping step and interpolation. Mean: 0.0146 pixels.

The out-of-plane velocity expresses the three-dimensional effects of the flow field: The intensity of the particles that move slightly out of the plane fades, and if the particle completely leaves the plane the gray value of the particle disappears.

Table 4.1 shows that some parameters were varied while others were kept constant: For image pairs 01, 04 and 05, for example, the number of particles is 1,000, 4,000 and 10,000, while all other parameters are fixed. The image pairs 01, 02 and 03 differ only with respect to the magnitude of the flow field: the average velocities are 2.5, 7.4 and 22 pixels/frame, but the flow field structure is the same in all three cases.

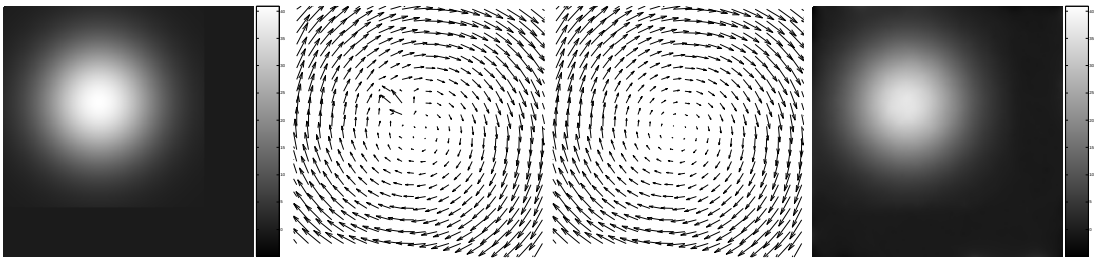

Figure 4.10.: Simple Rotation (case B): Same vector field as in case A, additional brightness (Gaussian shape) added. From **left** to **right**: Gaussian illumination pattern, reconstructed velocity field when applying Horn&Schunck approach (RMS error = 0.254 px), reconstructed velocity field when applying Horn&Schunck with additional brightness change handling term (RMS error = 0.017 px), reconstructed illumination pattern.

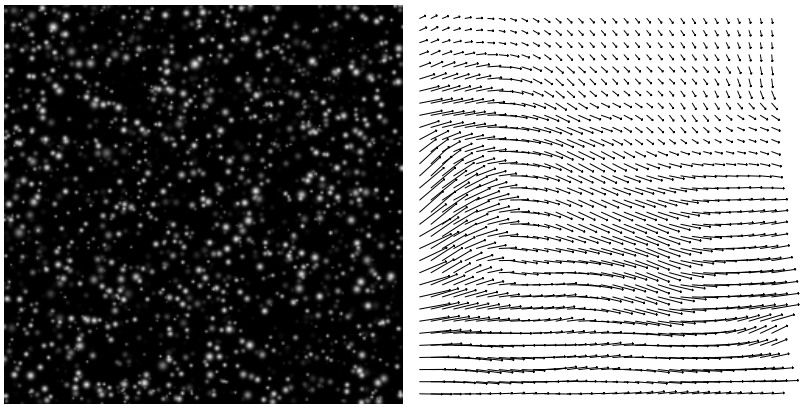

Figure 4.11.: Sample VSJ Standard Image with target velocity field

- **Synthetic combustion image pair:** Experimental conditions sometimes forbid to seed a fluid with particles. In some combustion processes, e.g., no particles can be brought into the fluid without interfering with the flow itself. Sometimes, however, the flow itself has a gray value structure that can be tracked. Figure 4.12 shows a synthetic example of such a structure. Traditional cross-correlation methods have problems with such data - they produce pronounced correlation peaks only in the presence of individual particles. Note that a second difficulty in these cases is, that - due to experimental conditions - there are local illumination changes from one frame to the next. As we are not considering physically motivated priors in this section, no care was taken to mimic a realistic velocity field. The motion that was added to the images is a simple sinusoidal vector field (cf. fig. 4.13) that has a maximum displacement of ≈ 2 pixels.

- **Real-world images:** We also included three real-world image pairs into our data set. Figure 4.12 shows a corresponding image from the first test case of a time-resolved PIV measurement of periodical vortices in the transitional cylinder wake [BLW03, WBL03]. The mean displacement is about 9 pixels/frame and the maximum displacement about 16

Image	Av. Displacement	Max. Displacement	Av. Out of Plane Vel.	Number of Particles	Av. Particle Diameter d	Std. Deviation of d
01	7.4 px	15.0 px	0.017	4000	5.0 px	1.4 px
02	**22.0 px**	**45.00 px**	**0.058**	4000	5.0 px	1.4 px
03	**2.5 px**	**5.1 px**	**0.006**	4000	5.0 px	1.4 px
04	7.4 px	15.0 px	0.017	**10000**	5.0 px	1.4 px
05	7.4 px	15.0 px	0.017	**1000**	5.0 px	1.4 px
06	7.4 px	15.0 px	0.017	4000	5.0 px	**0.0 px**
07	7.4 px	15.0 px	0.017	4000	**10.0 px**	**4.0 px**
08	7.4 px	15.0 px	**0.170**	4000	5.0 px	1.4 px

Table 4.1.: Pre-generated VSJ standard images. Variations from the default settings are marked in bold type.

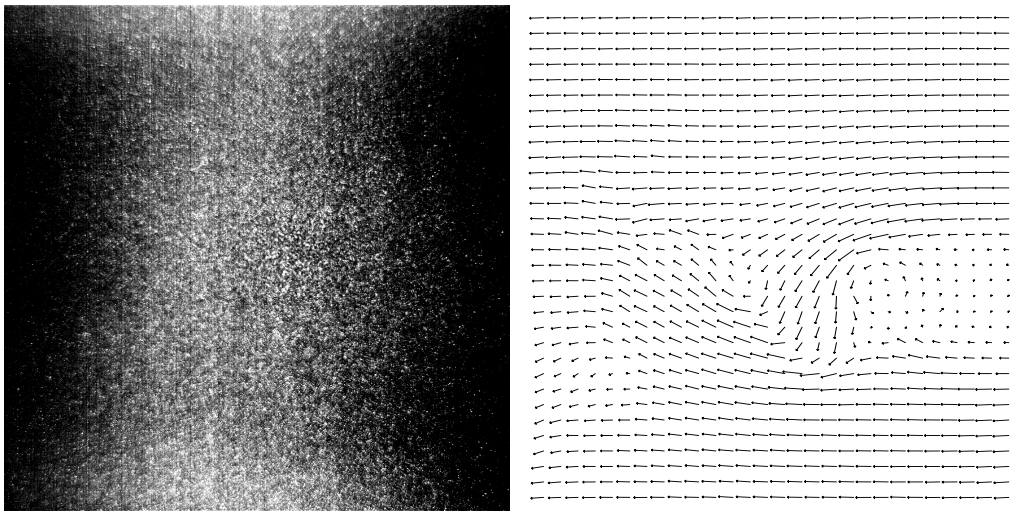

Figure 4.12.: Real-world image with estimated velocity field (variational approach)

pixels/frame. Figure 4.20 shows the second real-world test case: freezing in a convection box filled with water [QP98]. The mean displacement in this case is about 4 pixels/frame and the maximum about 15 pixels/frame. The third test case (cf. fig. 4.19) shows the wake behind a cylinder (size: 512×512 pixels, max. displacement ≈ 14 pixels) [Bur]. A special camera is used (Weinberger speedcam), whose sensor is divided into sixteen rectangular segments. The brightness of each segment is automatically adapted, so that a particle changes its brightness when it travels from one segment into the next.

Approaches and Parameter Settings

The data sets described above were evaluated using the following approaches and parameter settings:

- **Variational approach:** The spatial (∇I) and temporal ($\partial_t I$) derivatives were estimated using Derivative of Gaussian filters of size five at every point in the image domain. At the image borders (where the filter mask hangs over the image) the image is mirrored about its edge pixel (for smoothing operations) or reflected and inverted (for derivative operations).

  In a first series of experiments (**H&S R+S**), a setup of five resolution levels and nine

scale space levels on every resolution level was chosen. We set the smoothness parameter λ to $5 \cdot 10^{-3}$. However, we did not employ the additional brightness fitting term. In a second series of experiments (**H&S R+S+B**), we added this brightness fitting term, and set $\mu = 10$. For the VSJ and the real-world image pairs, only H&S R+S+B computations were performed. The parameters λ, μ were determined experimentally. However, we will show that, up to a certain point, changes of λ do not deteriorate results distinctly and that one can even improve the results by adapting λ manually. The gray values were scaled in each case to the interval $[0, 1]$.

- **DPIV approach:** For comparison we took the error measures of the classical 2D FFT based digital particle image velocimetry (DPIV) method from [QP98] in the synthetic test cases. Two different interrogation window sizes were applied: 32×32 pixel (**DPIV 32**) and 48×48 pixel (**DPIV 48**). We analyzed the "cylinder wake" real-world image pair using a hierarchical DPIV approach, with an interrogation window size beginning with 512×512 pixels and ending up with 64×64 pixels, with window-shifting and peak-height validation (but without substitution or interpolation, as we want to compare the actually computed values).

- **ODP2 approach [Qué92]:** We also considered the results of a dynamic programming-based optical flow technique. This approach transforms the two-dimensional correspondence problem to a sequence of one-dimensional search problems. It has been successfully applied to particle image velocimetry in [QP98]. The error measures were taken from [Qué99].

- **KLT approach [Che03]:** We also considered the results of a feature-tracking approach to motion estimation. The Kanade-Lucas-Tomasi Tracker tracks local areas of sufficient intensity variation; outliers are erased and a dense motion field is interpolated.

- **Integrated Continuity Equation [CHA+05]:** The authors use the integrated continuity equation (cf. sec. 4.1.1) together with a first-order regularization.

Note that the error measures of the competing approaches that we did not implement (i.e., DPIV, ODP2, KLT, ICE) were taken from the respective publications.

Error Measures

As quantitative error measures we computed the angular error (between correct and computed motion vectors) as defined in [BFB94] along with its standard deviation as well as the mean velocity error (L_1 norm of the difference between the correct and the computed velocities in pixels/frame).
For the Quénot image pair, the error measure was computed for the whole image except for the inner circular regions corresponding to the cylinders. Since the VSJ standard image pairs have different average velocities, the relative L_1 norm error (absolute error divided by average in-plane velocity) was computed in the corresponding series of experiments for the sake of comparability.

Numerical Results and Discussion

Simple Rotation Figure 4.9 shows our result (using the H&S R+S approach) for case A. In order to show the need for a back-warping of the resulting velocity field (cf. sec. 4.1.2), we estimate also the displacement error when assuming that $u_1(x_1 - \frac{1}{2}u_1, x_2 - \frac{1}{2}u_2) = u_1(x_1, x_2)$ (and resp. for u_2). This decreases the error by one order of magnitude. Figure 4.10 motivates the use of an additional brightness correction term. Without the term, a large error can be spotted at the location where the illumination changes. This error vanishes if we use the additional brightness correction term. Note that even the illumination pattern is reconstructed.

Quénot Image Pair Table 4.2 summarizes the error measures and their standard deviation ($\pm$) [5]. Furthermore, typical execution times of the respective algorithms are indicated.

		DPIV32	DPIV48	ODP2	KLT	H&S R	H&S R+S	H&S R+S+B
Perfect	angle	5.95 ± 13.9	9.35 ± 18.3	1.23 ± 2.24		1.36 ± 3.30	0.57 ± 1.85	0.58 ± 1.92
	disp	0.55 ± 0.94	0.87 ± 1.46	0.13 ± 0.10	0.50 ± 0.80	0.58 ± 1.67	0.06 ± 0.09	0.06 ± 0.09
Noise 5%	angle	6.49 ± 14.6	9.69 ± 19.0	1.83 ± 3.84		1.82 ± 3.66	0.77 ± 1.95	0.77 ± 1.98
	disp	0.61 ± 1.18	0.86 ± 1.49	0.21 ± 0.46	0.30 ± 0.80	0.76 ± 1.88	0.09 ± 0.09	0.08 ± 0.09
Noise 10%	angle	8.75 ± 17.9	10.8 ± 20.0	4.01 ± 10.8		2.50 ± 4.22	1.15 ± 2.25	1.12 ± 2.25
	disp	0.77 ± 1.57	0.91 ± 1.59	0.53 ± 1.44	0.31 ± 0.60	0.98 ± 2.12	0.13 ± 0.10	0.13 ± 0.10
Noise 20%	angle	35.0 ± 35.5	31.0 ± 30.4	6.70 ± 11.8		4.92 ± 5.67	2.16 ± 3.37	2.09 ± 3.28
	disp	3.11 ± 4.14	2.06 ± 2.88	0.88 ± 1.58	0.42 ± 0.60	1.93 ± 2.76	0.25 ± 0.19	0.25 ± 0.24
Add/rm 10%	angle	5.94 ± 13.5	9.52 ± 18.5	2.61 ± 9.94		1.58 ± 3.53	0.61 ± 1.88	0.62 ± 1.92
	disp	0.55 ± 0.93	0.87 ± 1.47	0.34 ± 1.28		0.72 ± 1.86	0.07 ± 0.09	0.07 ± 0.09
Add/rm 20%	angle	6.11 ± 14.2	9.77 ± 19.2	1.42 ± 2.54		2.72 ± 4.62	0.77 ± 2.06	0.76 ± 2.09
	disp	0.56 ± 0.99	0.88 ± 1.52	0.16 ± 0.12		1.36 ± 2.56	0.09 ± 0.10	0.08 ± 0.10
Mixed 5%	angle	6.40 ± 14.4	9.59 ± 19.0	1.77 ± 2.87		1.56 ± 3.39	0.81 ± 1.98	0.80 ± 2.01
	disp	0.60 ± 1.12	0.86 ± 1.51	0.20 ± 0.13		0.60 ± 1.67	0.09 ± 0.09	0.09 ± 0.09
Mixed 10%	angle	10.2 ± 19.6	11.3 ± 20.8	4.30 ± 11.7		1.99 ± 3.64	1.22 ± 2.33	1.18 ± 2.32
	disp	0.91 ± 1.89	0.93 ± 1.66	0.57 ± 1.71		0.98 ± 2.12	0.14 ± 0.10	0.13 ± 0.10
Mixed 20%	angle	40.8 ± 34.5	38.3 ± 29.7	6.15 ± 9.01		3.29 ± 4.61	2.41 ± 3.72	2.36 ± 3.73
	disp	3.73 ± 4.39	2.49 ± 3.19	0.74 ± 0.52		1.02 ± 1.98	0.31 ± 0.51	0.32 ± 0.58
Time		10 min	10 min	20 min	15 sec	16sec / 2sec	2 min / 15 sec	4 min

Table 4.2.: Quénot Image Pair: Angular error and absolute displacement error. H&S R+S and H&S R+S+B give very similar results (see text) and have a clearly better performance than DPIV and ODP2.

Note that DPIV and KLT yield sparse vector fields, whereas both ODP2 and the variational approach compute dense vector fields. All of the tested algorithms are rather insensitive to particle appearance/disappearance. However, they all are (in varying degrees) sensitive to superimposed noise. In the case of DPIV, extending the interrogation window size increases the robustness to noise, while decreasing the accuracy at the same time. However, irrespective of the window size, the performance of DPIV is much worse than the performance of the other approaches.
Comparing H&S R and H&S R+S, we realize that H&S R+S provides much better results in all the test cases. This had to be expected because temporal aliasing as well as linearization errors due to eqn. (3.7) are suppressed by additional scale space computations.

H&S R+S+B gives nearly the identical results as H&S R+S. This is clear as the considered synthetic image pairs do neither contain out-of-plane velocity components, nor do they model illumination changes.

[5] Error measures for the three algorithms not implemented by the authors were taken from [QP98], the execution times from [Che03].

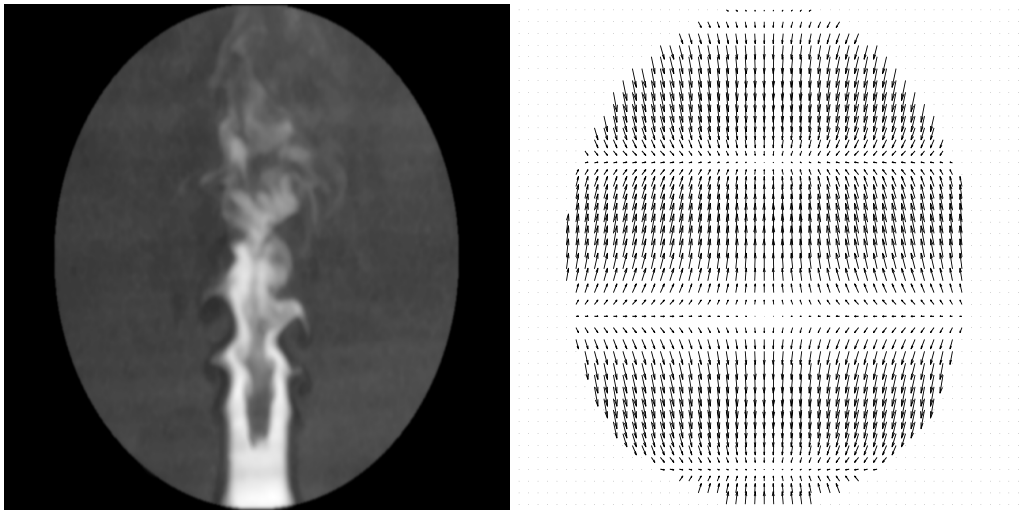

Figure 4.13.: Synthetic combustion image with target velocity field

Figure 4.14 shows the results for the "Perfect" test case. One can see that the highest estimation errors are reached at the borders of the two cylinders. The smoothness term penalizes the discontinuities at these locations and smooths over the discontinuities. The error at regions close to the left cylinder is the highest because of the high velocity of the fluid. The oval structures that can be spotted on the error and vorticity plots are probably due to the CFD grid that the authors used [Lu96].

The two lower images of figure 4.14 compare the exact vorticity field and the estimated vorticity field using our approach. With exception of the addressed problem (flow discontinuity at the left cylinder), the estimated vorticity field resembles the exact vorticity field very well.

H&S R+S and H&S R+S+B provide the best results in all test cases. The error measures of KLT are consistently better than those of H&S R, but slightly worse than the results of ODP2. However, it seems to be less noise-sensitive than ODP2, and has the advantage of much faster execution times than the computationally expensive dynamic programming technique. Note that ODP2 also seems to be more noise-sensitive than our variational approaches: While the error of the ODP2 approach is approximately twice as high as the H&S R+S approach's error for the "perfect" test case, its accuracy further decreases in the presence of noise: For a noise level of 20% ODP's performance is approx. three times worse than our variational method's performance, in the mixed 20% case, the factor is approx. four.

When we use a preconditioned conjugate gradient method to solve the H&S system matrices, the execution time of our algorithm is about 16 sec for H&S R, 2 min for H&S R+S and 4 min for H&S R+S+B (when choosing a residual error of 10^{-4} as a stopping criterion). Using a multi-grid approach (cf. sec. A.5) to solve the linear systems, the computation time of H&S R is approx. 2 sec, while the H&S R+S computation takes about 15 sec on an up-to-date computer. Information about the different multi-grid cycles and stopping criteria can be taken from [BWF+05]. Thus our approach is as fast as the feature tracker and faster than ODP2. Even real-time operation can be achieved through parallelization using domain decomposition [KSBW04]. Note, however,

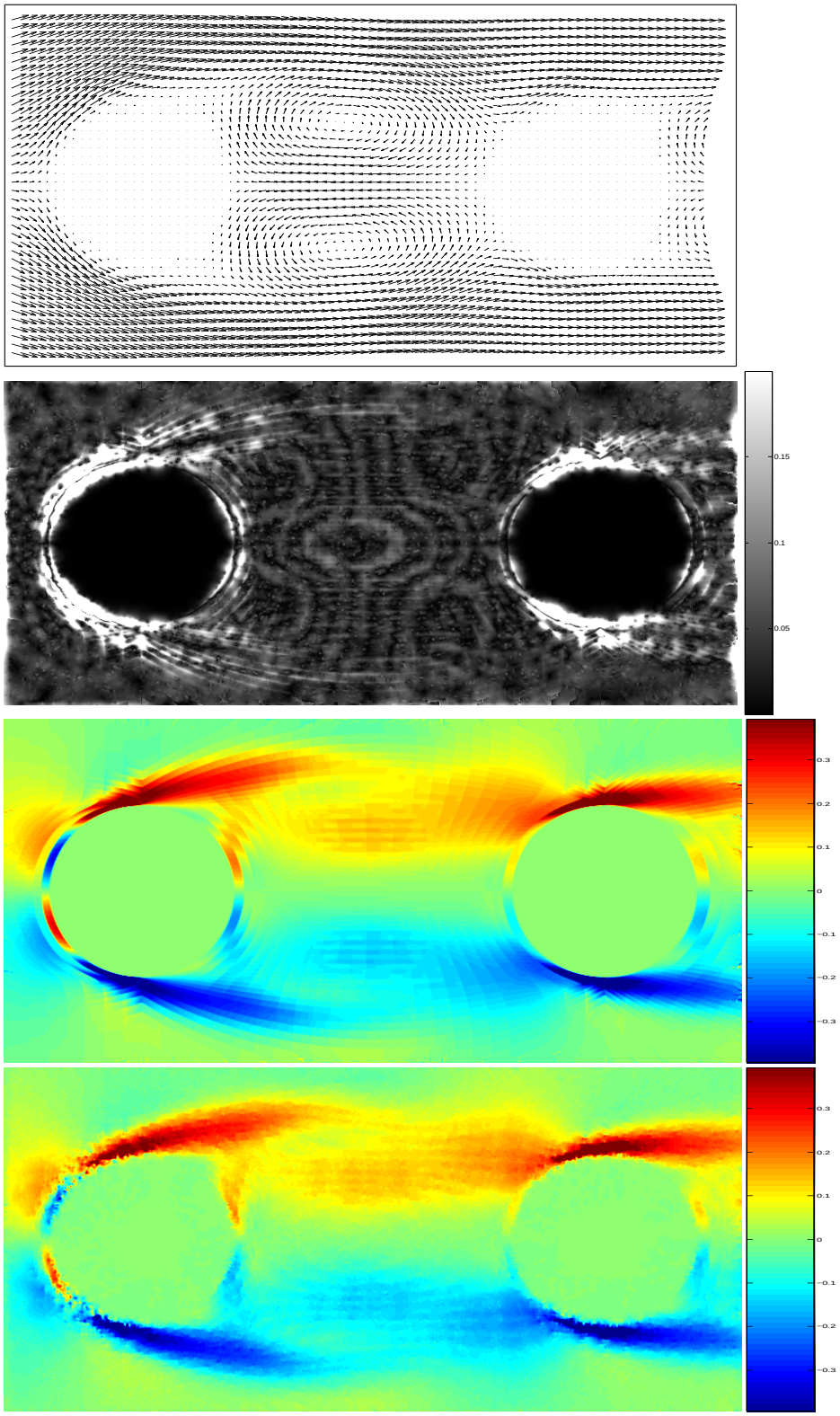

Figure 4.14.: Results for the Quénot image pair “Perfect”. Estimated flow field with H&S R+S (top), absolute displacement error (2nd row), exact vorticity field (3rd row) and estimated vorticity field (bottom).

that, while we use a 3 GHz Intel processor, the DPIV and ODP results were obtained using much older and slower machines. Quénot mentions in [QP98] a 250 MHz SGI processor and a 200 MHz Intel processor.

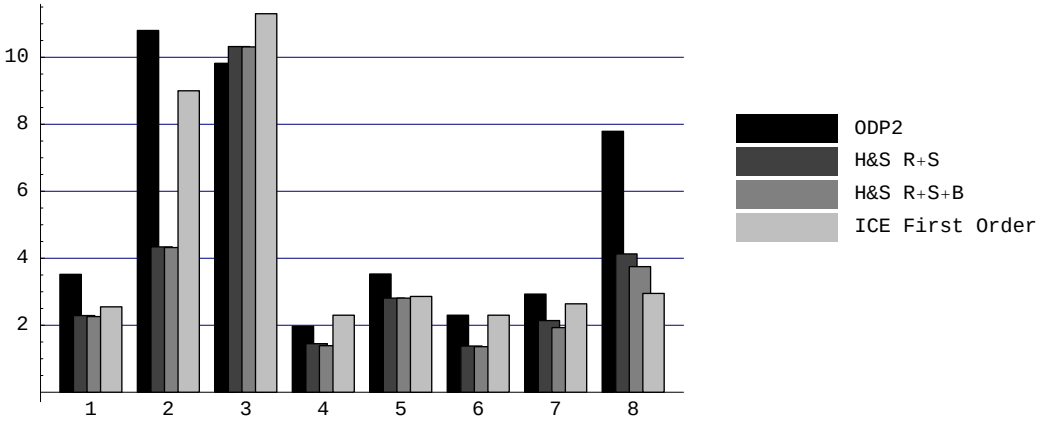

Figure 4.15.: Average relative L_1 error of the different optical flow approaches for the VSJ standard image pairs 01-08.

VSJ Standard Image Pairs Figure 4.15 compares the average L_1 errors of all mentioned optical flow-based approaches. The errors of the H&S R+S+B approach are constantly slightly smaller than the errors of the H&S R+S approach. A reason is that out-of-plane velocity components and image noise were modeled in the generation of the synthetic images.

The average relative L_1 error of the H&S R+S+B computation for image pairs 01 and 04-07 is constantly between 1.40 (image pair 04 has a high particle density) and 2.79 (image pair 05 has a low particle density). As a consequence, the number of particles seems to be the parameter of the image that influences most the quality of the flow field estimation.
Throughout the VSJ standard image pairs, the error measures of all considered approaches are more or less the same: H&S R+S+B tends to exhibit the best performance, followed by H&S R+S and ICE. ODP2 exhibits a slightly worse performance than the other techniques. The reason for ICE not being superior to the much simpler optical flow constraint is probably the lacking connection between image brightness and fluid density. In PIV imagery (contrary to meteorological images where extremely good results have been achieved using ICE), the fluid itself is usually untextured (and therefore invisible) and only the individual particles can be spotted. The luminance of these particles, however, is not influenced by the surrounding fluid's density.

H&S R+S+B gives only insignificantly better results than H&S R+S because the standard images do not model spatio-temporal illumination changes.

Figure 4.16 shows that our approach is rather insensitive to changes of the smoothness parameter λ. However, if we adapted this parameter manually for every image pair, we could achieve even better error measures than the ones shown in figure 4.15.

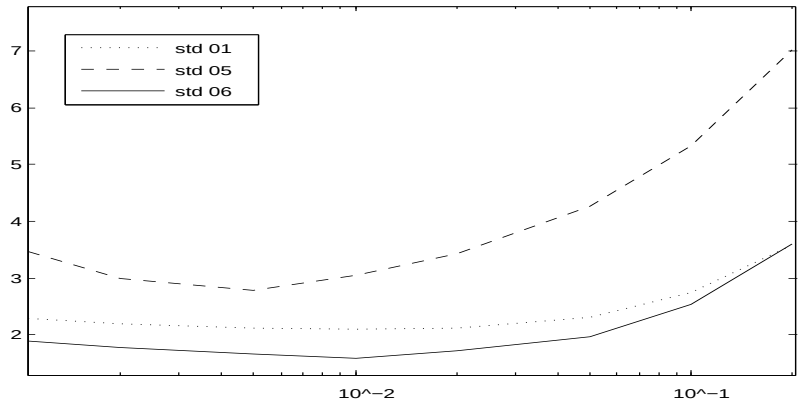

Figure 4.16.: Average relative L_1 error (in %) of H&S R+S+B for the VSJ standard image pairs 01, 05, and 06 for changing smoothness parameters λ (logarithmic ordinate).

Synthetic combustion image pair This image pair makes the need for the additional illumination computation clear. Without the additional brightness correction term, the estimated velocity field is extremely inaccurate: The absolute L1 error is 0.585 pixel. With the term, the error drops to 0.083 pixels - the estimated velocity field (cf. fig. 4.13) resembles the target velocity field very well.

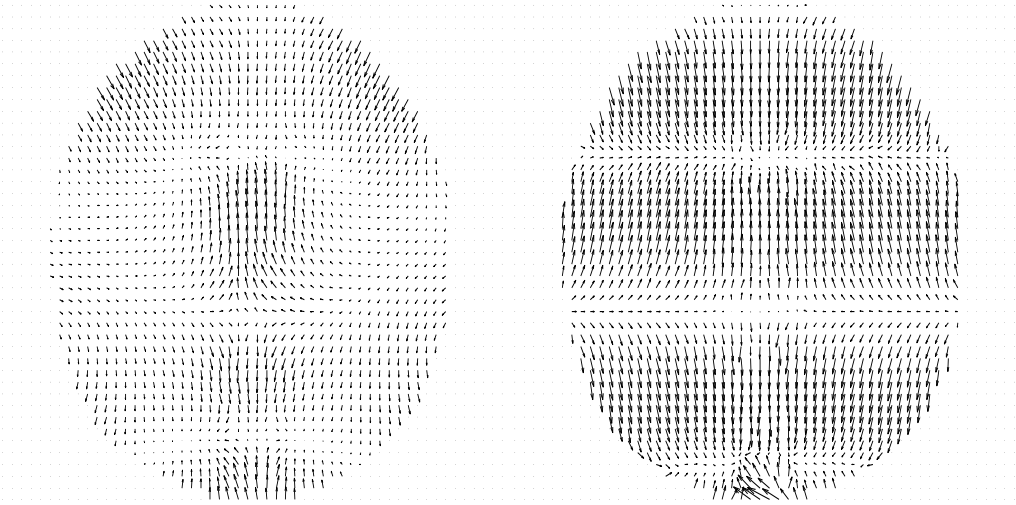

Figure 4.17.: Combustion image pair: Estimated velocity fields. **Left:** H&S R+S. Abs. L1 error = 0.585 pixels. **Right:** H&S R+S+B. Abs. L1 error = 0.083 pixels.

Results for Real-World Image Pairs Figure 4.18 shows the results for the first real-world image pair ("cylinder wake") computed with the variational approach and DPIV.

One can clearly see that the variational approach resembles the true motion field much better than the cross-correlation approach. At regions with abruptly changing motion (i.e. the turbulence emerging behind the cylinder in the middle of the image), the DPIV method is not able

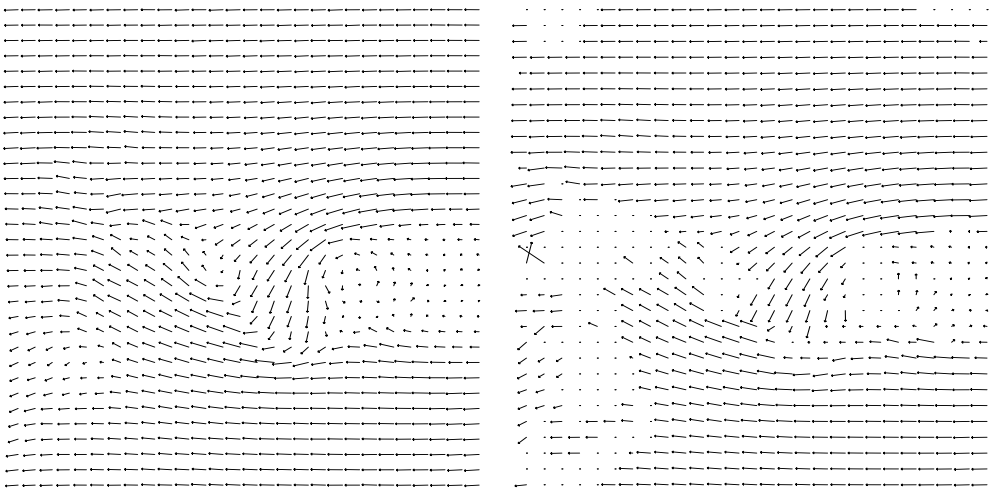

Figure 4.18.: Cylinder Wake: Results. **Left:** Dense vector field computed with the variational approach. **Right:** Sparse vector field computed with DPIV.

to accurately determine the velocity field. This is mainly due to the limited spatial resolution, which leads to a violation of the assumption of a constant velocity inside interrogation windows at these locations. The statistical character of correlation-based processing, however, prohibits the use of smaller interrogation windows. Furthermore, in regions dominated by out-of-plane velocities (i.e. at the left border of the image), the cross correlation approach fails as well: Since no global velocity information is used, the probability of outliers is markedly increased at these locations, hence a valid flow field cannot be computed.
Figure 4.20 compares the H&S R+S results of the "freezing" image sequence with the results that Quénot achieved with ODP2. Both results resemble the true motion field very well. With the exception of the borders (where the gray value is constant and therefore no reliable motion can be estimated) and one spot in the middle of the image (where the velocity is high and varies locally very strongly), the absolute L_1 difference is persistently smaller than 0.5 pixels. From the visual impression, however, it is impossible to tell which of the two estimates is more precise.

Figure 4.19 shows the results for the real-world image pair with changing brightness values from one segment to the next. While H&S R+S fails to recover the true velocity field (due to the brightness fluctuations from one segment to the next), H&S R+S+B is able to recover it quite accurately.

4.1.5. Conclusion

We have successfully modified a prototypical variational optical flow estimation approach for the purpose of Particle Image Velocimetry. The novel approach outperforms the standard cross correlation methods and computes dense motion fields.

A decisive advantage of the variational approach (4.3) is its potential for further development.

Various extensions of the simple smoothness term in (4.3) are possible, such as spatio-temporal regularization [WS01b], div-curl-shear regularization [Sch94] or non-quadratic discontinuity-preserving regularization [WS01a], for instance.

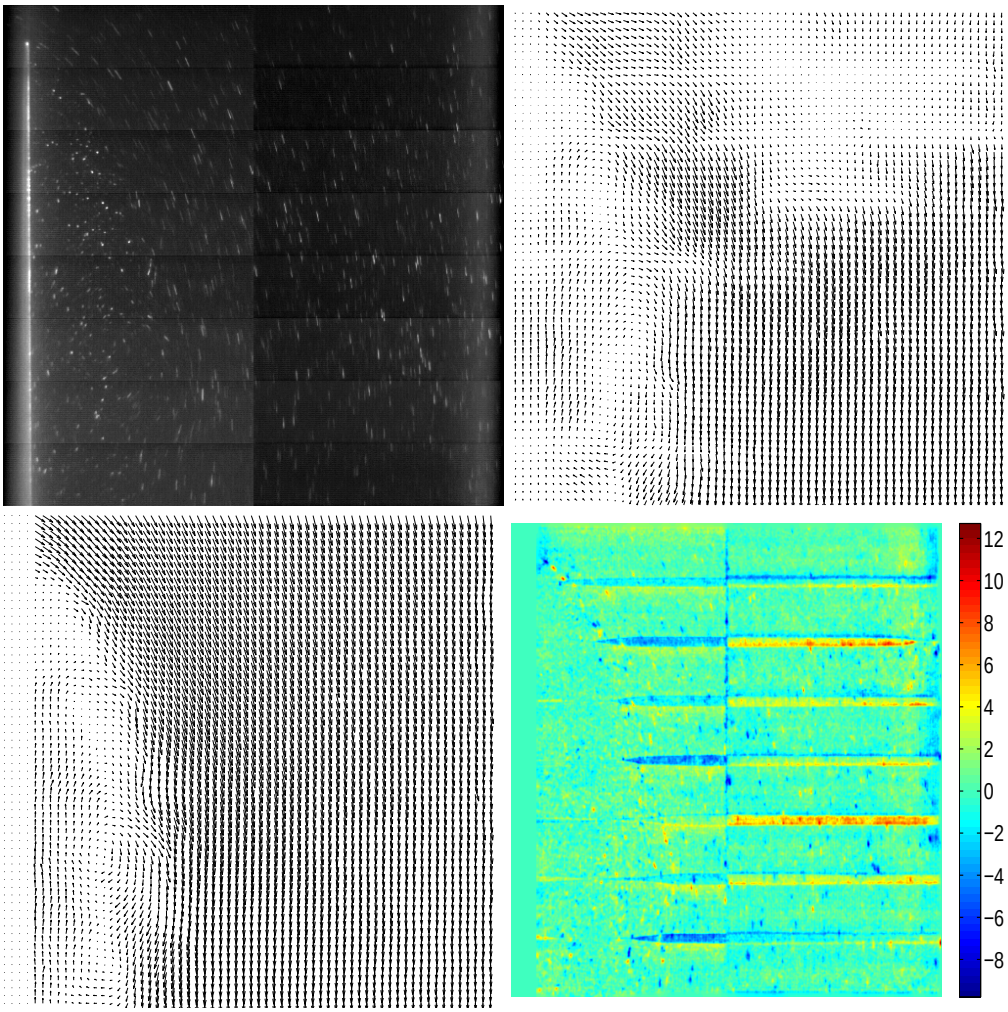

Figure 4.19.: **Top Left:** Real-world image pair (cylinder wake). Note the brightness changes from one segment to the next. **Top Right:** Due to the changing brightness, H&S R+S is not able to reconstruct a valid velocity field. **Bottom Left:** H&S R+S+B is able to recover the velocity field. **Bottom Right:** Recovered brightness field (estimated brightness change from one frame to the next in %). The transitions between the segments are recovered reasonably well.

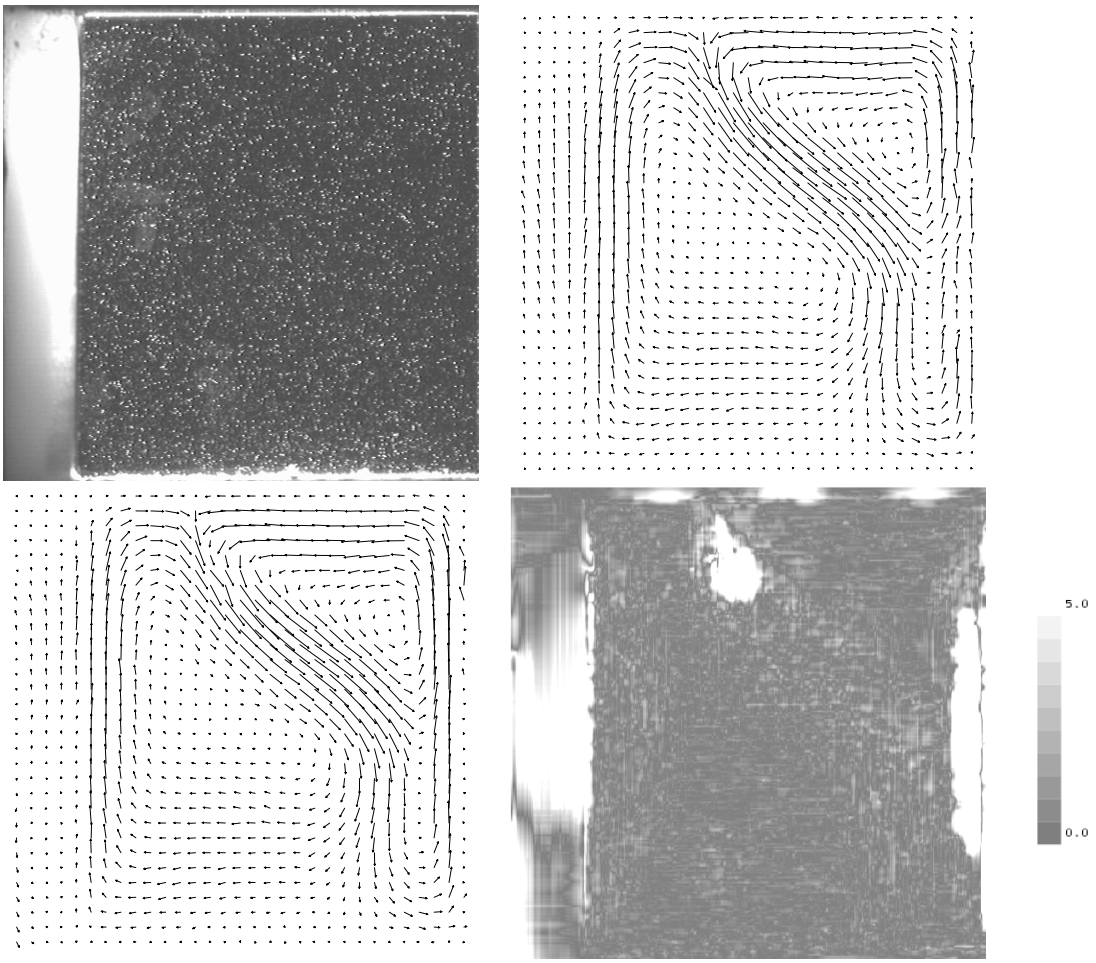

Figure 4.20.: Freezing in a differentially heated cavity. **Top row:** Particle image and computed velocity field (H&S R+S). **Bottom row:** Velocity field computed with ODP2 (left), comparison of the absolute displacement values of the two solutions (right).

We believe, however, that the most important advantage of this type of variational method for PIV is the possibility to include *physically motivated* prior knowledge: The governing equations of fluid mechanics are PDEs, and our variational framework allows the incorporation of these PDEs. In chapter 5 we will exchange the simple first-order regularization term by higher-order regularizers and will subsequently incorporate physics-based priors into our framework.

4.2. Variational Particle Tracking Velocimetry

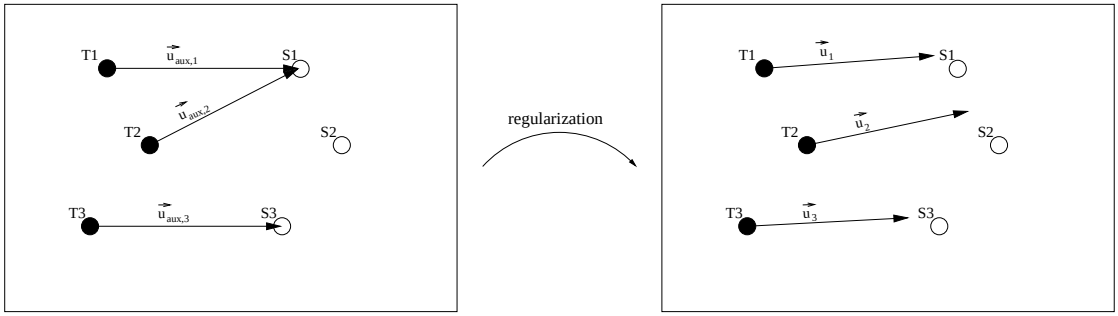

Figure 4.21.: Black circles denote particle positions in the first frame, white circles denote positions in the second frame. **Left:** Simple nearest-neighbor search yields mismatches. **Right:** Nearest-neighbor search followed by regularization with smoothness constraint. In the next iteration, T2 will find the correct match.

The objective of this section is to generalize the class of variational approaches to Particle *Tracking* Velocimetry. To this end, we have to replace the *continuous* data term of variational approaches to PIV, with a *discrete* non-differentiable particle matching term for PTV. This raises the problem of minimizing such data terms together with a *continuous* regularization term. We accomplish this with an advanced mathematical method, which guarantees convergence to a local minimum of such a non-convex variational approach to PTV.

Figure 4.21 illustrates the basic behavior of this new type of variational approach to PTV. On the left, figure 4.21 depicts a common situation where particle matching by nearest-neighbor search fails. The variational PTV-approach presented in this paper is able to avoid, and even to revise, such erroneous local decision through the smoothness term (figure 4.21, right). A key advantage in our opinion is that all “rules” guiding the matching of particles are encoded by the choice of a smoothness term which, in turn, can be related to physical properties of the underlying fluid, like low divergence for example [YRMS05]. The physical constraints are thus incorporated *directly* (in contrast to e.g. the indirect incorporation in genetic algorithm approaches, cf. sec. 2.2.2). In the following, we will introduce this novel variational approach to PTV (section 4.2.1) and the corresponding optimization procedure (section 4.2.2). The investigation of different smoothness terms in this context is left for future work. Numerical experiments for benchmark image pairs, a comparison with three alternative approaches, as well as results for real-world image sequences will be presented in section 4.2.3. We conclude in section 4.2.4 by indicating various extensions to this prototypical approach within the variational framework.

4.2.1. General Problem Formulation

Basic Assumptions and Constraints

Let S denote the coordinates of the extracted particles in the first image of an image pair, and T denote the coordinates of the extracted particles in the second image. Then, we define the distance of a specific particle with coordinates S^i to T by

$$d_T(S^i) := d(S^i, T) = \inf_{T^i \in T} d(S^i, T^i),$$

where $d(S^i, T^i)$ is just the Euclidean distance. Therefore, the target velocity field u (where u^i denotes the displacement of particle S^i from frame 1 to frame 2) minimizes the accumulated distance function

$$D(u) = \sum_{i=1}^{M} d_T(S^i + u^i) \ , \tag{4.6}$$

where $u = u^1, u^2, ..., u^M$, and where M is the number of extracted particles in image 1.

Unfortunately, minimization of (4.6) is a highly non-convex problem, as *every* other possible matching minimizes the equation as well. The *local* minimum is just the "nearest-neighbor" solution. We define a convex attraction potential as an increasing continuous function that attracts every particle to its closest neighbor:

$$E_{local}(u) = \sum_{i=1}^{M} \frac{\alpha}{2} \big(d_T(S^i + u^i)\big)^2 \ . \tag{4.7}$$

Up to this point, the particles are only attracted to their nearest neighbors and the minimization of (4.7) is trivial. This is why we have to make an additional assumption about u. The prototypical assumption that we want to make use of in this paper, is the assumption of *smoothness*. We indicate in section 4.2.4 that other assumptions (that include e.g. physical knowledge) are conceivable.
However, rather than considering vector fields that are close to constant in a small region (the predominant assumption in PTV), we want to rule out too irregular vector fields by minimizing the magnitudes of the spatial (and, in case of image sequences, spatio-temporal) gradients of u:

$$E_{global}(u) = \int_\Omega \sum_{j=1}^{N} |\nabla u_j(s)|^2 ds. \tag{4.8}$$

Please note that $u = (u_1, u_2, ..., u_N)^\top$, where N indicates the dimensionality of the problem (N is usually 2 or 3). The integration variable s is for image pairs in 2D $s = \big(x_1, x_2\big)^\top$, and in 3D $s = \big(x_1, x_2, x_3\big)^\top$, where x_1, x_2 and x_3 denote the spatial coordinates within the domain Ω. For image sequences follows $s = (x_1, x_2, t)^\top$ in 2D, and $s = (x_1, x_2, x_3, t)$ in 3D, where x_1,x_2 and x_3 denote the spatial coordinates, and t the temporal coordinate.
Equations (4.7) and (4.8) can be combined into the variational framework

$$E(u) = E_{loc.}(u) + \lambda E_{glob.}(u) = \underbrace{\sum_{i=1}^{M} \frac{\alpha}{2} \big(d_T(S^i + u^i)\big)^2}_{\text{data}} + \lambda \underbrace{\int_\Omega \sum_{j=1}^{N} |\nabla u_j(s)|^2 ds}_{\text{regularization}}, \tag{4.9}$$

where E_{local} is called "data term" which incorporates *local* information, and E_{global} is the *global* regularization term. In this work, the so-called smoothness parameter $\lambda \geq 0$ is considered a user parameter that controls the smoothness of the resulting velocity field. If we choose $\lambda = 0$, no regularization is performed. The reconstructed velocity field is therefore just the "nearest neighbor" solution, as the *locally* optimal solution for every particle in image 1 is the matching with its nearest neighbor in image 2.

Outlier Treatment

An important problem in the PTV analysis is raised by the fact, that usually not all the particles are detected correctly. In 2D it may happen that a particle is visible in the first frame, but moves out of the illuminated plane and is therefore not visible, or beneath the threshold, in the second frame. In 3D, additional problems occur when the 3D reconstruction fails, e.g. due to a very high particle density. Further problems arise from particle images tending to coalesce.

We can distinguish between two error scenarios:

- A particle is extracted from the second image, but not from the first image: In this case the proposed algorithm can still estimate a reliable velocity field, as it searches matches for all particles in the first frame.
- A particle is visible only in the first frame but not in the second frame: In this error case, the nearest-neighbor search (4.11) of the proposed algorithm will necessarily find the wrong match in *every* iteration (cf. figure 4.22). Through the smoothness term of (4.12) this error is propagated to the neighborhood of the erroneous vector.

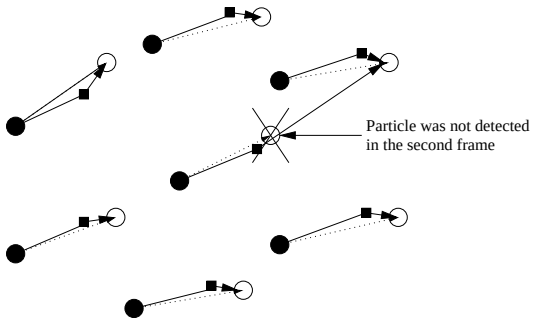

Figure 4.22.: Black circles denote particle position in the first frame, white circles denote positions in the second frame. Filled rectangles denote the current estimate. One particle has not been detected in the second frame. Minimization of (4.11) necessarily leads to the wrong match.

The strategy that we want to take, is to eliminate vectors that contribute a high energy to (4.12). This is achieved through a threshold: We replace the attraction potential of the data term of (4.11) by a robust potential - a cut-off potential that cuts off points located beyond an adjustable threshold. These outliers are not considered in the regularization step of the current iteration.
However, the *result* of the regularization step *is* propagated to the outliers: Linear interpolation yields the velocity field also at the locations of the outliers, the positions of which are updated,

as are the positions of the inliers. The idea is that they may be torn below the threshold in case they were wrongly detected as outliers.

In order to improve the performance in image regions with high velocity, we start with a low outlier threshold and then slowly increase this threshold: Thus, in the first iterations, particles in fast moving regions will tend to be considered outliers, while particles in slowly moving regions will tend to be considered inliers. The idea is that, in the course of several iterations with an attenuating threshold, more and more particles will be considered inliers and the estimated velocity field in the high-velocity regions can converge to the correct flow field. Extensive experiments have confirmed this behavior.

4.2.2. Optimization and Discretization

Note that the implicit data constraint defined by equation (4.6) is a non-convex function. Thus, retrieving a local minimum of (4.9) does not imply having found the global optimum.

We use an auxiliary variable approach that represents a sound mathematical framework and guarantees convergence [Coh96]: In a two-stage iterative algorithm, each iteration is composed of a local deformation followed by a global regularization. To justify this approach we modify the energy $E(u)$ of (4.9) by introducing an auxiliary variable u_{aux}. The two above steps can then be interpreted as alternate minimizations with respect to each of the two variables, the variable of the initial energy u and the auxiliary variable u_{aux}.

A general formulation of the energy E_{aux} following [Coh96] and based on formula (4.9), with the extra auxiliary variable $u_{aux} = u^1_{aux}, u^2_{aux}, ..., u^M_{aux}$, has the form:

$$\begin{aligned} E_{aux}(u, u_{aux}) \quad &= \sum_{i=1}^{M} \left(\tfrac{1-\alpha}{2} \big(d_{S+u}(S + u^i_{aux})\big)^2 + \tfrac{\alpha}{2} \big(d_T(S + u^i_{aux})\big)^2 \right) \\ &\quad + \lambda \int_\Omega \sum_{j=1}^{N} |\nabla u_j(s)|^2 ds \; . \end{aligned} \tag{4.10}$$

The first two terms of equation (4.10) exhibit the auxiliary variable's role as an interpolate between $S + u$ and T. Globally, we can think of the iterative minimization of E_{aux} as a deformation of the current vector field followed by a regularization. The successive minimization of E_{aux} is equal to subsequent minimization of the following two energies E_I and E_{II}, each with respect to a different variable - E_I with respect to u_{aux}, and E_{II} with respect to u:
Local deformation:

$$E_I(u_{aux}) = \sum_{i=1}^{M} \left(\frac{1-\alpha}{2} \big(d_{S+u}(S + u^i_{aux})\big)^2 + \frac{\alpha}{2} \big(d_T(S + u^i_{aux})\big)^2 \right) \tag{4.11}$$

Global regularization:

$$E_{II}(u) = \sum_{i=1}^{M} \left(\frac{1}{2} \big(d_{S+u}(S + u^i_{aux})\big)^2 \right) + \lambda \int_\Omega \sum_{j=1}^{N} |\nabla u_j(s)|^2 ds \; . \tag{4.12}$$

The two equations can be subsequently iterated in the given order until convergence is obtained. Equations (4.11) and (4.12) demonstrate how both minimizations are linked by the term $\sum_{i=1}^{M} \big(d_{S+u}(S+u^i_{aux})\big)^2$. The minimizing u_{aux} of E_I can be interpreted as a trade-off between the closeness to $S + u$ and the closeness to T. This gives a good direction of displacement and

avoids too large deformations of the auxiliary flow field u_{aux}. The grid generation is performed using a Delaunay triangulation [Aur91] (cf. fig. 4.23). For technical details about grid generation and discretization of eqn. (4.12), for 2D and 3D image pairs and image sequences, we refer to appendix A.3 and [RGS04].

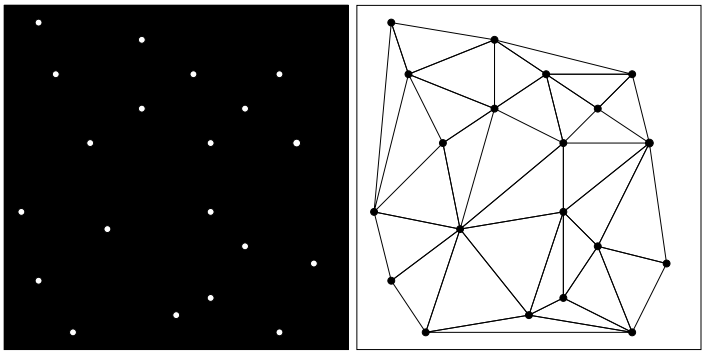

Figure 4.23.: Delaunay triangulation of the area covered by particles from an image plane. Line intersections denote extracted particle positions.

4.2.3. Experimental Evaluation

In this section, we test the variational PTV approach on synthetical and real 2D and 3D data sets. For the 2D case, we report comparisons of our variational particle tracking approach with three other approaches. Before discussing the results, we first describe the data sets used for the comparison, the preprocessing (i.e. particle extraction and 3D reconstruction), the alternative approaches and their corresponding parameter settings, and the quantitative error measures.

Data

The experimental evaluation was carried out on the basis of the following data sets:

- **Synthetic Data:** The Visual Society of Japan (VSJ) has published standard images for particle image velocimetry that are freely available on the Internet [ONK00a]. For 2D data, we will refer to the test image classified as 301 in the VSJ library. It consists of 10 frames taken in intervals of 0.005 sec; each frame consists of about 4,150 particles. It shows the vertical portion of the impinging jet, with a maximum velocity of 10 pixels/ frame. Figure 4.24 shows the first image from this series along with the correct motion field. We will analyze our 3D approach using the test images classified as 331 in the VSJ library (jet shear flow). Figure 4.25 shows a plot of its velocity field.
  The advantage of the VSJ images is that the underlying motion fields, as well as the particle coordinates, are available so that the evaluation of different approaches, as well as that of different parameter constellations, is possible. By basing our computations on this particle position data, we have to deal with very high particle concentrations (approx. 4,150 particles to be tracked in the 2D case and 3,500 particles in the 3D case). We want to evaluate the performance of our algorithm in cases of high particle concentrations, as up-to-date CCD cameras yield increasingly high resolutions, and thus an up-to-date tracking

system must be capable of managing high particle concentrations.
In order to achieve a more realistic test scenario we will randomly delete particles in order to emulate typical individual particle extraction errors.

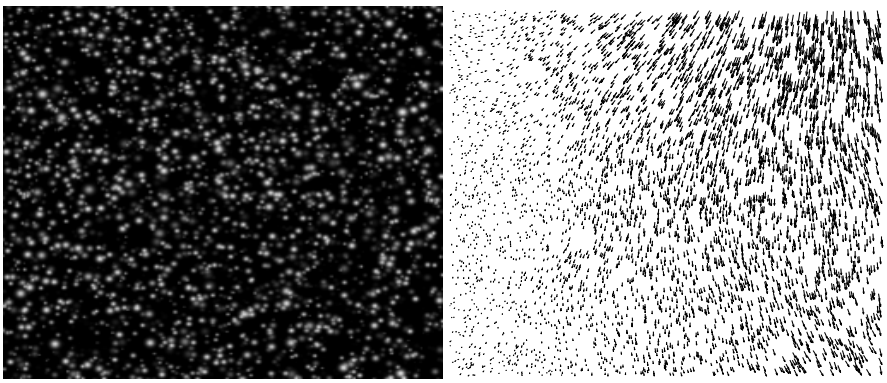

Figure 4.24.: VSJ Standard Image 301.

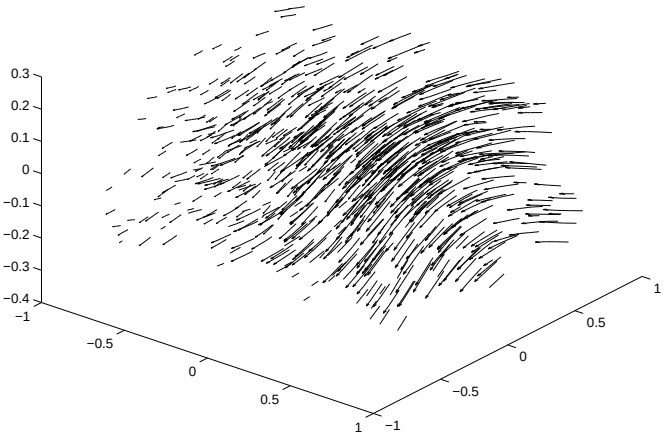

Figure 4.25.: 3D velocity field of VSJ Standard Image 331.

- **Real-World Data:** Figure 4.26 shows different frames from a visualized air flow. The used camera is a high-speed camera with 1000 fps. The flow is visualized by Styrofoam particles or micro balloons. One has to visualize the flow in a way that the mean displacement is approximately 10 px./ frame.
  To analyze the 3D capability of our algorithm, we took the “stirred aquarium” sequence from [Maa92a]. It investigates the water flow in a channel made of glass. The velocity of the real flow in the glass channel averaged 30-50 cm/s. The camera system, which contains 3 video cameras that operate with 25 fps, was moved continuously in direction of the flow to optimize the tracking. In order to get the characteristic flow, one has to consider the bias of the moved camera system.
  For successful processing, a compromise between camera frame rate, exposure time, flow

velocity and illumination is necessary. Fast particles detected when using a longer exposure time appear as streaks. This means a loss of accuracy and it generates ambiguities. Longer exposure times also reduce the maximum frame rate. Low frame rates in turn increase errors due to the curvature of the particle paths. High frame rates are correlated with short exposure times, which results in dim particles. Especially particles at the boundaries of the illumination corridor cannot be segmented by an overall threshold.
In order to get sufficient results in terms of a successful tracking, the highest possible frame rate is required. The more turbulent the flow is, the higher the time resolution has to be in order to get correct matching outputs. The maximum time delay between two epochs depends on the feasibility of the temporal matching. The latter in turn depends on the homogeneity, the turbulence, and the velocity of the flow and, of course, on the performance of the matching algorithm and its ability to incorporate spatio-temporal constraints on homogeneity.

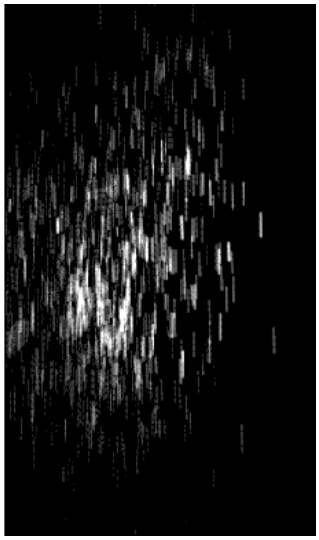

Figure 4.26.: Real-World Image: 4 frames have been superimposed to visualize the overall motion.

Preprocessing Steps

Individual-Particle Detection In order to track individual particles, these particles first have to be extracted out of the gray-value structure of the image. Many authors have concentrated on this topic (a comparison of different particle detection approaches can be found in [OL00]).
While we omitted the particle detection and 3D reconstruction steps in the synthetic cases by directly basing the tracking algorithm on the provided 3D coordinates, we used the so-called particle mask correlation method described in [ET99, SSKH00] in the 2D real-world cases. For the 3D real-world case, a region-growing approach with a discontinuity parameter that divides overlapping particles was used [Maa92b].

3D Reconstruction For 3D PTV, the illuminated scene is usually recorded by a system of three or more cameras. If the camera calibration is known, then, by use of epipolar geometry, the particle positions can be described by a set of lines. The particles are situated around the intersection points of these lines. According to the accuracy of the calibration and the particle density there will be ambiguities. In some cases it is not possible to resolve them [Maa92a]. As the presented algorithm is capable of handling outliers (cf. sec. 4.2.1), we will, in these cases, consider all *possible* particle locations that cannot be ruled out - anticipating that the wrong candidates will be considered as outliers by the algorithm.

Approaches and Parameter Settings

The data sets described above are evaluated with the use of the following approaches and parameter settings.

- **Variational approach (VAR):** The particle coordinates are normalized so that all particles lie between 0 and 1 in all spatial dimensions; the temporal dimension is numbered in integer steps ($t = 1, 2, ..., T$).
  For all the test cases we use a smoothness parameter of $\lambda = 0.1$. The parameter α is set to 0.8. In the first iteration, 75% of the particles are considered as outliers and in every iteration, 0.1% particles in addition are considered as inliers. No additional particles are considered as inliers if the outlier threshold reaches 0.01.
  The iteration is stopped if no further decrease in energy occurs.

- **Four-frame in-line tracking method (FIT) [HC91, KSS89]:** The movement of the tracer particles is traced frame by frame while the geometrical consistency of every possible particle path is checked. Therefore an iterative procedure of, firstly, the extrapolation of the particle displacement, and secondly, the search for the nearest neighbor is implemented. As this method asks for four consecutive frames to track the particles, we will use all of the four frames of the VSJ Standard Images.

- **Binary-image cross-correlation method (BCC) [UYO89]:** This method is considered a variation of the standard cross-correlation PIV, in which the correlation functions are computed for each interrogation window which is centered on the first-frame particles. An adaptive shifting scheme is used.

- **Relaxation method (NRX) [BL96, OL00]:** This analysis is based on the probability of particle matching between the first and second frames, defined for every possible pair of particles, including the probability of there being no match. A high probability of matching is assumed if the neighboring particles move similarly.

Error Measures

In this work, we want to concentrate on two error measures: *yield* and *reliability*.

- Yield (E_Y) is the measure of the number of correct vectors produced between two images (n), divided by the total number of particle pairs known to exist between the two images (v):

$$E_Y = \frac{n}{v} \, .$$

- Reliability (E_R) is the measure of the number of correct vectors that were reconstructed by the tracking method (n), divided by the total number of vectors determined by the tracking method (d):

$$E_R = \frac{n}{d} \, .$$

It is apparent that we can influence both error measures by the chosen parameters: If we use a high outlier threshold, we can expect a good reliability (as only matches that fit the model very well are considered), while E_Y will definitely drop. A lower threshold will lead to an increase in E_Y, while decreasing the reliability.

Numerical Results and Discussion

2D Results The first test case is the computation of the velocity field between the frames 0 and 1 of the VSJ 301 image sequence. After 700 iterations the solution presented in figure 4.27 (outlier ratio: 3%) is generated.

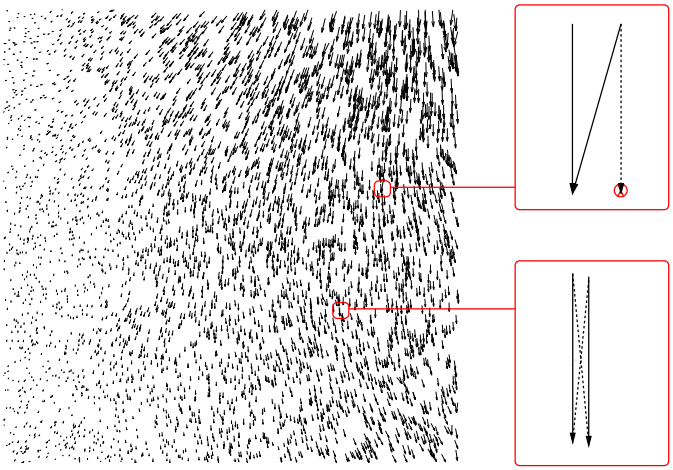

Figure 4.27.: **Left:** Estimated velocity field VSJ Image 301. **Right:** Two likely error constellations: One particle has not been extracted in frame 2, the matching is performed with a close neighbor of this vanished particle **(top)**. Due to three-dimensionality of the velocity field, two particles "cross" in the two-dimensional projection. The two-dimensional variational approach presumes smoothness of the projection and chooses the wrong match **(bottom)**.

In the test case, $4,042$ particles are visible in both images; $4,039$ matchings are computed; $3,894$ of which are correct. This corresponds with a yield rate of $E_y = 96.34\%$ and a reliability rate of $E_R = 95.93\%$. Figure 4.28 shows these two error measures through the iteration process. The average angular error of our estimated vectors is $0.24°$, and the root mean squared (RMS) error is 0.0261 px., which suggests that the performance of our approach is much more exact than that of (cross-correlation based) PIV techniques [6]. However, these numbers are mislead-

[6] Multi pass cross correlation (Davis 7.1.1.34), e.g., yields an RMS error of 0.0742 px. for the frames 0 and 1 of the VSJ 301 image sequence.

ing: When using highly accurate matching techniques, the overall RMS or angular error will be largely caused by inaccuracies in particle extraction.

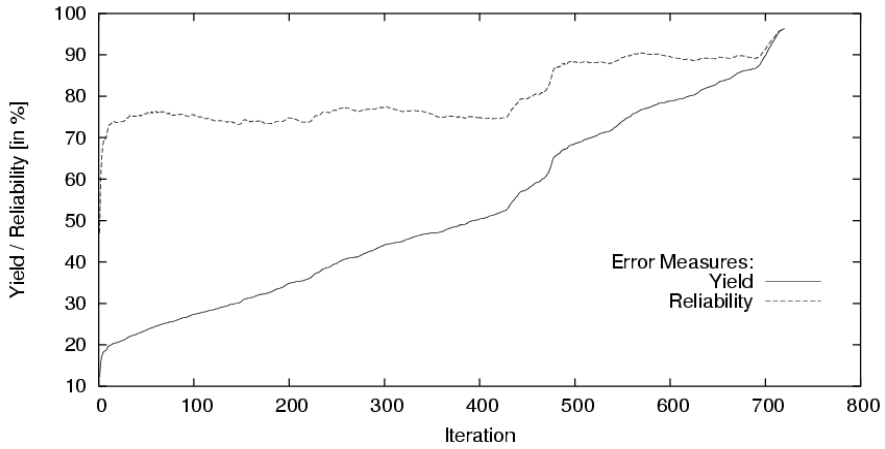

Figure 4.28.: Graph of the two error measures from iteration 1 to 720.

Figure 4.30 points out our outlier strategy. As explained in sec. 4.2.1, we start using a very low outlier threshold, considering only 25% of the particles as inliers. These particles are located mainly in the left part of the image (top left figure). The velocity is small and the velocity field is smooth, so that these particles fit the model best. In the course of iterations, more and more particles are considered as inliers, so that the reconstructed velocity field resembles the true velocity field even at locations where the velocity is very high. Beginning with iteration 400, more and more particles in the lower right part of the image are considered as inliers and find their correct counterparts (cf. figure 4.30). The velocity induced by these correct matchings, in turn, is propagated through the smoothness constraint to the middle of the image where it raises the number of correct matches. This is why reliability strongly increases between iterations 450 and 500 (cf. figure 4.28).
Figure 4.29 shows that our approach is rather insensitive to changes of the smoothness parameter λ. However, as λ goes to zero, the reconstructed velocity field approaches the "nearest neighbor solution" (cf. sec. 4.2.1) and the accuracy drops. In contrast, if we apply a very large smoothness parameter, the algorithm will not be able to deal with spatial and temporal motion variation, many particles will be considered as outliers, and the performance will decrease.
Please note that we used *the same* λ in *all* our experiments - if we had adapted the parameter manually for every experiment we could have achieved better results than the ones presented in this paper. Changes of the parameter α had only very little influence on the resulting matching.

Table 4.3 compares the results achieved with our variational approach, with the results of the approaches introduced in sec. 4.2.3. In order to guarantee a fair comparison we have not used the correct particle coordinates provided by the VSJ, but extracted particle positions by using the particle mask correlation method (cf. sec. 4.2.3). Therefore the amount of particles is clearly lower than in the preceding computations. The variational PTV method finds the largest amount of matches while additionally yielding the highest reliability.

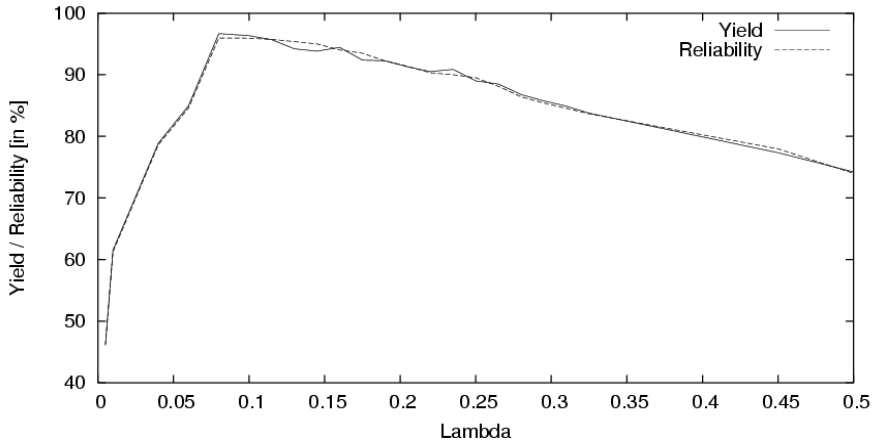

Figure 4.29.: Graph of the two error measures for different smoothness parameters.

Algorithm	Frames	Estimated Matches	Correct Matches	Reliability
FIT	0,1,2,3	630	559	88.73%
BCC	$0 \rightarrow 1$	860	788	91.62%
	$0 \rightarrow 2$	863	691	80.07%
NRX	$0 \rightarrow 1$	808	788	97.52%
	$0 \rightarrow 2$	714	680	95.24%
VAR	$0 \rightarrow 1$	872	865	99.20%
	$0 \rightarrow 2$	904	885	97.90%

Table 4.3.: Comparison of four PTV algorithms: Four-frame in-line tracking (**FIT**), Binary-image cross-correlation (**BCC**), Relaxation (**NRX**), Variational Approach (**VAR**).

2D + Time Results (2D Image Sequences) The next step is the additional exploitation of temporal smoothness information. Therefore we have to analyze the whole VSJ 301 image sequence consisting of 10 frames. Figure 4.31 shows the computed trajectories. Table 4.4 shows the parameters we use and the results that we achieve. Furthermore, the results of the analysis of image pairs only are indicated. In every frame, the computation based on the whole sequence is at least as good as the image pair result. This had to be expected, as additional information is available in the sequence case. The reason why only slight improvements are achieved has already been addressed: We analyze a 2D projection of a 3D velocity field, therefore the smoothness assumption does not necessarily hold at every point in the image. This is why we will later turn to three-dimensional problems.

Figure 4.32 shows the computed trajectories for the four frames of the 2D real-world image. Visual comparison of the extracted velocity field and the image pair suggests the assumption that no wrong match has taken place.

3D Results First we want to compute the 3D velocity field between the frames 0 and 1 of the VSJ 331 image sequence. The solution that was generated after 750 iterations is presented in figure 4.33. In this test case, $3,364$ particles are visible in both images and $3,372$ matchings are computed. These matches include all exact matches, and 8 particles that do not have a counterpart in the second image, but are erroneously matched to another particle. As expected, the 3D results are much better than the 2D results. Computations with volume coordinates of the

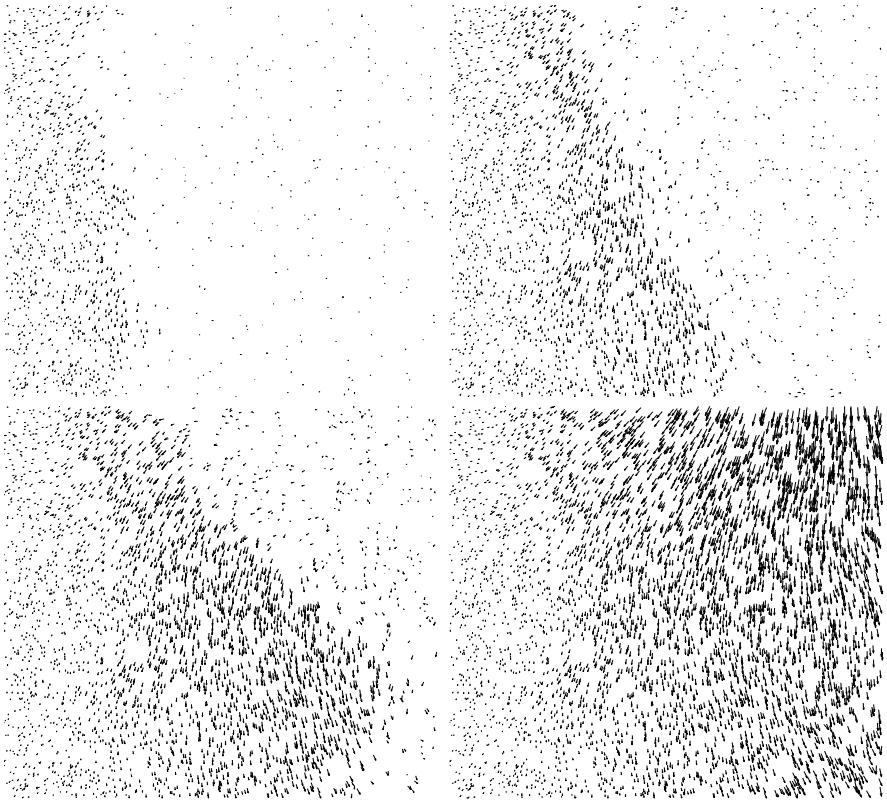

Figure 4.30.: Estimated velocity field after 50 **(top left)**, 300 **(top right)**, 500 **(bottom left)** and 720 **(bottom right)** iterations.

VSJ 301 sequence show that we achieve matching rates very close to 100% in these test cases as well.
Please note, however, that in real-world scenarios, errors in 3D imaging and 3D reconstruction will lead to missing particles or erroneous particle locations. This is why we want to test the robustness of our algorithm: Table 4.5 shows that even an increase in outlier probability does not deteriorate the results significantly. In these test cases, the indicated percentage of particles (first column) has been randomly removed from *both* images, to simulate problems in particle extraction and 3D reconstruction. The second column indicates the number of particles that are visible in both frames, columns three and four show the two performance measures.

In order to assess the limits of our approach we want to consider only every second image. The results indicated in Table 4.5 show that the error measures are still very good. When considering only every third image, however, the approach is no longer able to determine a valid velocity field. In fact, both yield and reliability drop to 0% (i.e. not a single velocity vector is recovered correctly). The algorithm does not find a starting point, as the offsets at *every* position in the image are so high that *no* particle is able to find its counterpart in the first iteration, and thus the algorithm converges to the wrong minimum. This drawback had to be expected as we are minimizing a highly non-convex functional (cf. eq. 4.11).
Even in this case the perfect matching can be found, if we provide the algorithm with a good initial guess (e.g. by specifying the overall image velocity, or one single match).

	$\alpha = 0.8, \lambda_{sp} = 0.1, \lambda_{tmp} = 10$		$\alpha = 0.8, \lambda = 0.1$	
Frames	Image Sequence		Image Pairs	
	Yield	Reliability	Yield	Reliability
00→01	97.72	96.41	96.34%	95.93%
01→02	97.62	96.61	96.83%	95.83%
02→03	97.69	96.45	97.05%	95.81%
03→04	97.64	96.64	96.90%	95.90%
04→05	97.32	96.63	97.05%	96.35%
05→06	97.64	96.70	97.11%	96.18%
06→07	97.28	96.10	92.99%	91.86%
07→08	97.28	96.30	93.11%	92.10%
08→09	96.33	95.24	93.94%	92.87%

Table 4.4.: Error measures for VSJ Standard Image 301.

	$00 \rightarrow 01$			$00 \rightarrow 02$		
Removed Particles	Possible Matches	Yield	Reliability	Possible Matches	Yield	Reliability
0%	3,364	100.00%	99.76%	3,192	99.97%	99.47%
5%	3,037	100.00%	99.84%	2,881	99.86%	99.45%
10%	2,731	100.00%	99.60%	2,586	99.38%	99.34%
15%	2,440	100.00%	99.59%	2,307	98.22%	99.60%
20%	2,170	100.00%	99.40%	2,053	98.30%	99.56%
25%	1,885	100.00%	99.74%	1,809	44.83%	44.81%
30%	1,649	100.00%	99.40%	1,557	38.79%	39.35%
35%	1,403	99.93%	99.64%	1,339	31.14%	31.17%
40%	1,211	100.00%	99.26%	1,131	32.98%	33.01%

Table 4.5.: Error measures for VSJ Standard Image 331.

Figure 4.34 shows the extracted trajectories from the real-world sequence “stirred aquarium” [Maa92a]. The sequence consists of 31 exposures of the whole volume with a three-camera setup. 1.300-1.400 particles were detected in every image. 3D reconstruction yielded a total number of 28.818 particles from the sequence (i.e. ≈ 930 particles / frame). We used the same parameters we used for the synthetic experiments. 22.485 matches were found when using our variational algorithm (i.e. ≈ 750 matches / image pair). The position of each vector is expressed in the initial camera coordinate system.

4.2.4. Conclusion

We have introduced and successfully evaluated a variational Particle Tracking Velocimetry approach that combines a discrete matching term and a continuous regularization term. This novel approach can handle 3D image sequences, and int outperforms standard PTV methods.

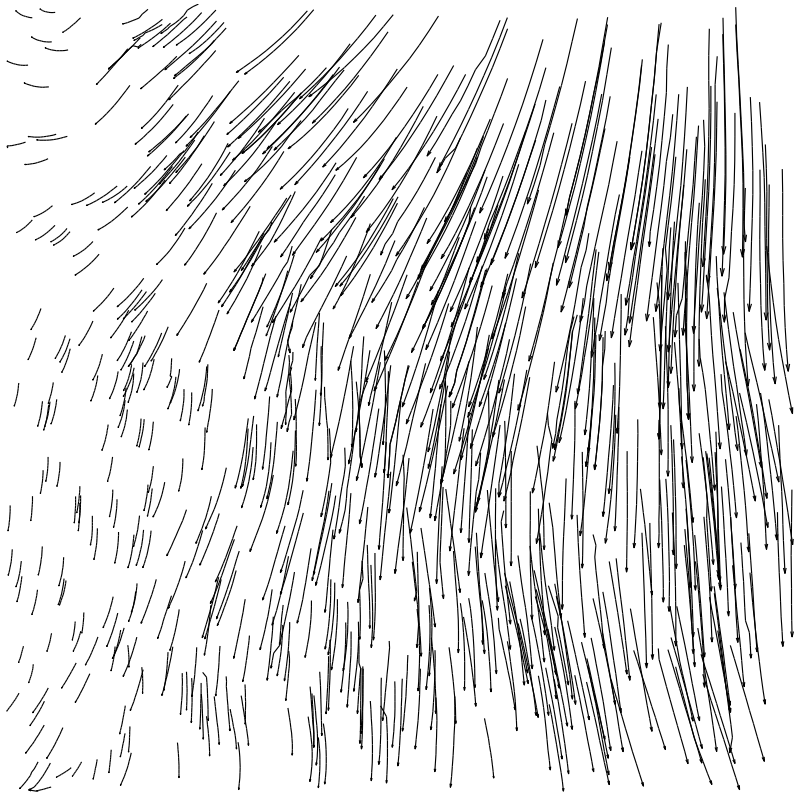

Figure 4.31.: Computed trajectories from sequence VSJ 301. For perceivability, only every tenth trajectory is plotted.

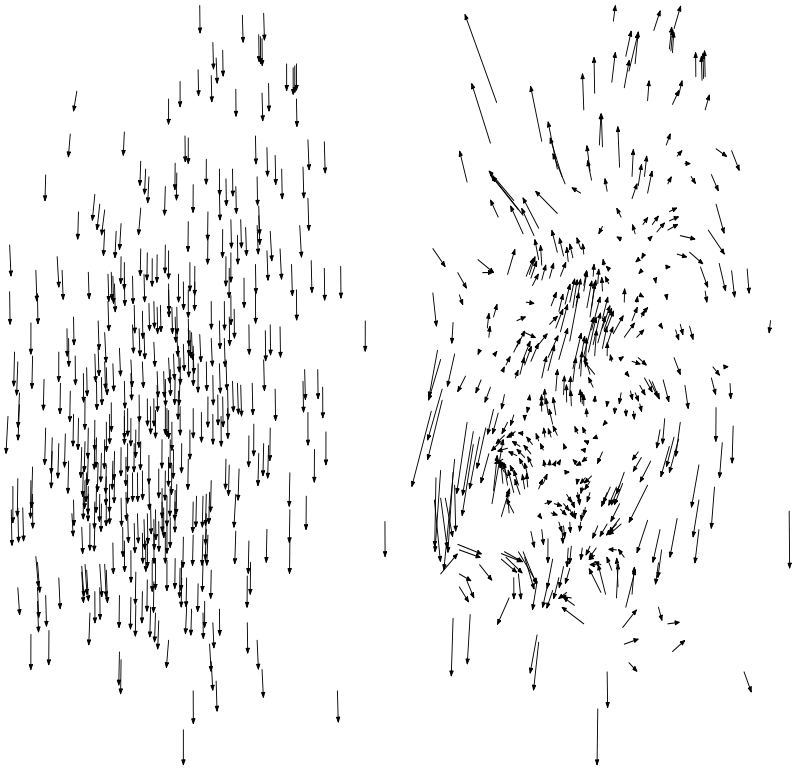

Figure 4.32.: **Left:** Computed trajectories from real-world image sequence. **Right:** Velocity vectors between frame one and frame four with mean flow component subtracted; amplified for perceivability.

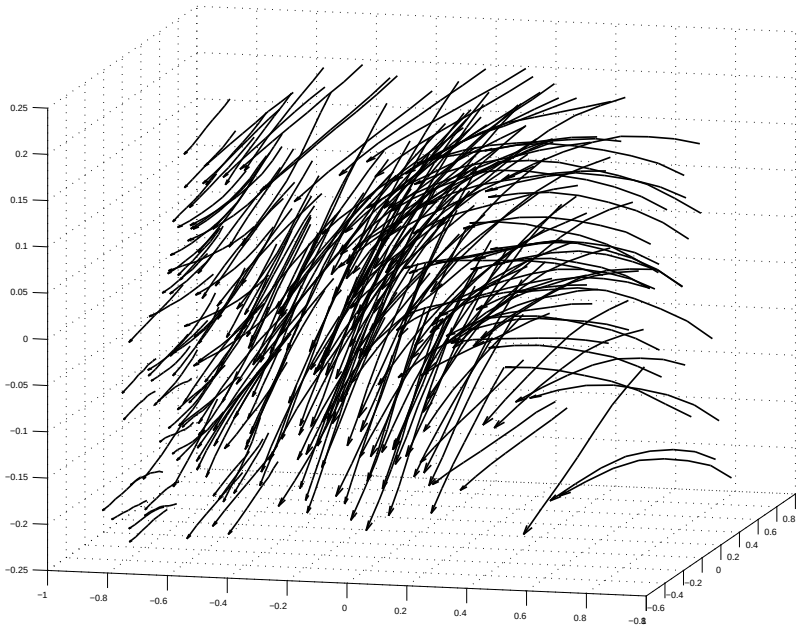

Figure 4.33.: Estimated 3D velocity field for sequence VSJ 331. For perceivability, only every tenth computed trajectory is plotted.

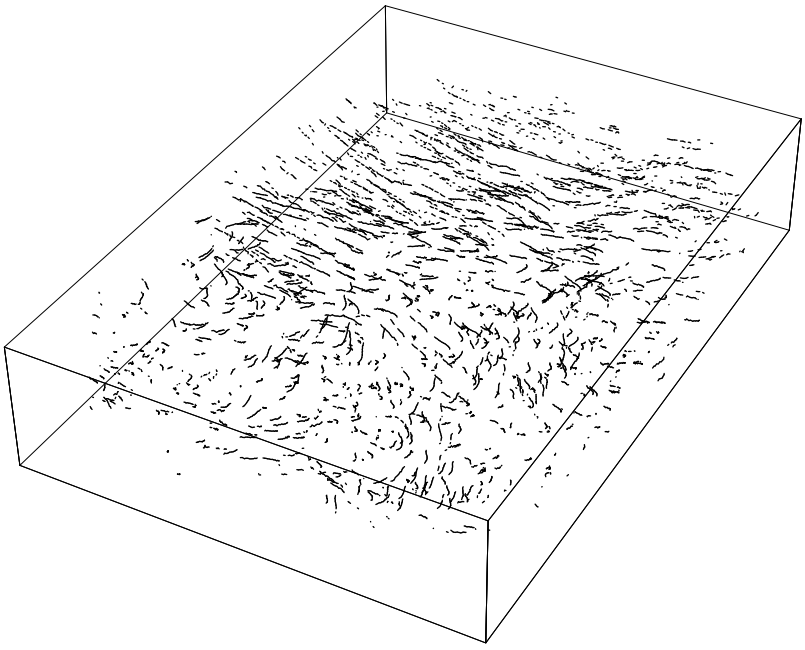

Figure 4.34.: Estimated 3D trajectories for the real-world sequence “stirred aquarium”. Every third trajectory has been printed.

4.3. PIV or PTV?

A decisive question that the reader might pose at this point is:

> *What is the better choice, PIV or PTV?*

Besides well-known advantages of PTV

- **High Resolution** (in comparison with CC-PIV): one velocity vector per particle,
- **3D Capacity:** In 3D, PTV can be supported and combined with stereoscopic analysis and 3D reconstruction, leading to high-resolution 3D3C vector fields,

we cannot answer this question satisfactorily: Suppose our variational PTV approach is used for temporal particle tracking (i.e. tracking from one frame to the next). Then, the main share of the overall velocity error will be caused by the individual particle extraction algorithm (and, for 3D PIV, by 3D reconstruction) and *not* by failed temporal matching. So, in order to answer the above question, one would first have to analyze typical error ranges of individual particle extraction methods. This is, however, out of the scope of this manuscript and thus left for further research.

5. Variational Fluid Motion Estimation: Physics-based Regularization

After we have adapted the data term of typical variational approaches for motion estimation to the requirements of PIV images in the preceding chapter, we will use the second part of this thesis to develop regularization terms that are better suited for typical fluid motion than the simple regularizer introduced in section 3.4. First, we review regularizers that were especially created for fluid image analysis. We will argue in section 5.2 that these methods, though they do yield very reasonable results, lack a sound physical interpretation. Consequently, we will introduce novel physically motivated regularizers:

- In sec. 5.3, we will propose the use of the so-called Stokes equation as prior knowledge when analyzing PIV image pairs.
- In sec. 5.4 the full Navier-Stokes equations (in the form of the vorticity transport equation) will be used as prior knowledge for the analysis of whole PIV sequences.

5.1. Higher-Order Regularization

5.1.1. First Order Div-Curl Regularization

Let us consider a 2D vector field $u = (u_1, u_2)^\top$ that is defined inside a domain Ω and is zero at its boundaries. The *Helmholtz decomposition* states that this vector field can be interpreted as a superposition of a divergence-free (*solenoidal*) vector field and a curl-free (*irrotational*) vector field:

$$u = u_{so} + u_{ir} \ , \tag{5.1}$$

with

$$\text{div } u_{so} = \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} = 0$$

and

$$\text{curl } u_{ir} = -\frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2} = 0 \ .$$

In case of non-zero boundary conditions, the decomposition comprises a *laminar* term that is both divergence- and curl-free:

$$u = u_{so} + u_{ir} + u_{lam} \ . \tag{5.2}$$

In the preceding chapter we have used the simple smoothness-term of Horn&Schunck for regularization. Using the Helmholtz decomposition, this smoothness term can be rewritten as

$$E(u) = \int_\Omega |\nabla u_1|^2 + |\nabla u_2|^2 dx = \int_\Omega (\text{div } u)^2 + (\text{curl } u)^2 dx. \tag{5.3}$$

if we demand a vanishing velocity field at the image boundaries.[1] Expression (5.3) can be expanded to control divergence and rotation of the velocity field *directly* and *independently* of each other

$$E(u) = \int_\Omega \alpha(\text{div } u)^2 + \beta(\text{curl } u)^2 dx \tag{5.4}$$

using two different regularization parameters $\alpha, \beta > 0$ [Sut93].[2] (5.3), however, illustrates also the main drawback of first-order regularization: The assumption of a smooth velocity field u is equivalent to the assumption of small divergence and curl components of u, which is usually invalid for real-world fluid flow.

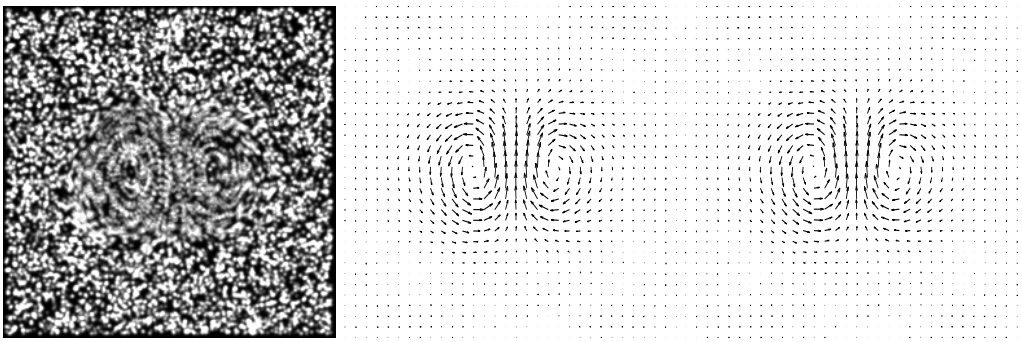

Figure 5.1.: **Left:** Synthetic Fluid Image Pair (both frames superimposed). **Middle:** Exact velocity field. **Right:** Estimated velocity field (Horn&Schunck R+S).

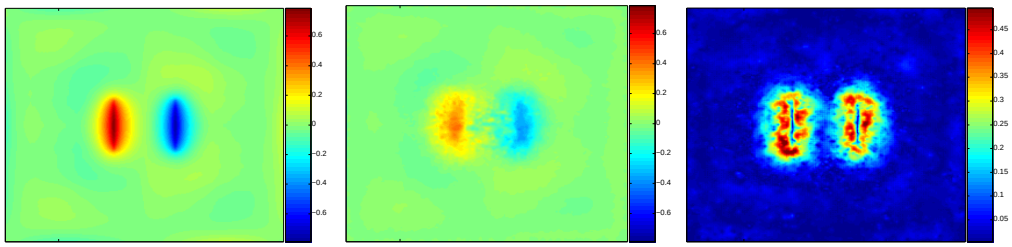

Figure 5.2.: **Left:** Exact curl of velocity field. **Middle:** Estimated curl (Horn&Schunck R+S). **Right:** L_1 Velocity Error: mean 0.056 px. (Horn&Schunck R+S).

[1] For arbitrary flows (i.e. flows with non-zero boundary values) an additional shear component is introduced and (5.3) can be rewritten as

$$E(u) = \int_\Omega |\nabla u_1|^2 + |\nabla u_2|^2 dx = \frac{1}{2}\int_\Omega (\text{div } u)^2 + (\text{curl } u)^2 + (\text{shr } u)^2 dx,$$

with shr $u = \sqrt{(\partial u_1/\partial x_1 - \partial u_2/\partial x_2)^2 + (\partial u_1/\partial x_2 + \partial u_2/\partial x_1)^2}$.
Note that the shear-component is often disregarded. This can be justified by either precomputing and subtracting the flow over the image boundaries or by artificially expanding the domain of computation beyond the image domain and assuming that u vanishes at infinity.

[2] Note that in fluid imagery, the divergence component will usually be much smaller than the rotational component, as the apparent velocity field is the 2D projection of an incompressible fluid. Therefore it will be advantageous to use a rather large regularization parameter α.

While the Horn&Schunck reconstruction of a typical velocity field of a fluid (cf. figure 5.1) appears reasonable, fig. 5.2 shows that the curl is severely underestimated. This yields errors at locations within the image, where the curl is large.

5.1.2. Higher Order Div-Curl Regularization

The standard way to attenuate the smoothing properties of the Horn&Schunck regularization term is the use of higher-order regularization. Due to the observation that there are mainly compact areas within the flow where divergence or curl are large (cf. 5.2, left), it makes sense not to demand smoothness of the velocity components but smoothness of divergence and curl [AB91, Sut94b, GP96]:

$$E(u) = \int_\Omega \alpha|\nabla \text{div } u|^2 + \beta|\nabla \text{curl } u|^2 dx. \tag{5.5}$$

While (5.5) tends to preserve divergence and curl, it is rather difficult to implement, as the corresponding Euler-Lagrange system consists of two coupled fourth-order PDEs. One possibility is to mollify the second-order constraint by introducing the auxiliary variables ξ and ω, which can be seen as approximations of the true divergence and curl, resp. [CMP02, CHA+05]:

$$E(u, \xi, \omega) = \int_\Omega \lambda(|\text{div } u - \xi|^2 + |\text{curl } u - \omega|^2) + \mu(||\nabla \xi||^2 + ||\nabla \omega||^2) dx, \tag{5.6}$$

where μ is a positive parameter. Note that (5.6) decouples the fourth-order PDE associated with (5.5) into a system of second-order PDEs. [CMP02] describes an iterative strategy of minimizing the optical flow constraint together with the regularizer (5.6). Basically the energy functional is minimized alternatively with respect to u, ξ and ω. Figure 5.3 shows the gain of this type of second-order regularization: The mean L_1 velocity error decreases, while the curl is much better reconstructed (cf. fig. 5.4).

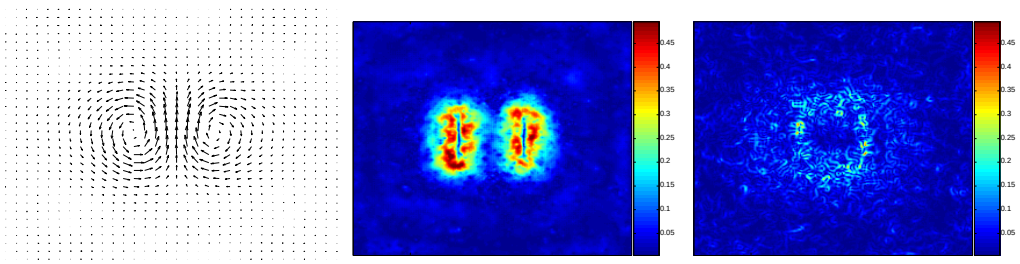

Figure 5.3.: **Left:** Estimated velocity field (second-order regularization). **Middle:** Velocity Error (Horn&Schunck R+S): mean 0.056 px. **Right:** Velocity Error (2nd-order reg.): mean 0.032 px.

5.1.3. Flow Decomposition Using Potential Functions

Let us come back to the Helmholtz decomposition that we have introduced in sec. 5.1.1. It is well-known that the solenoidal (u_{so}) and the irrotational components (u_{ir}) of the velocity field

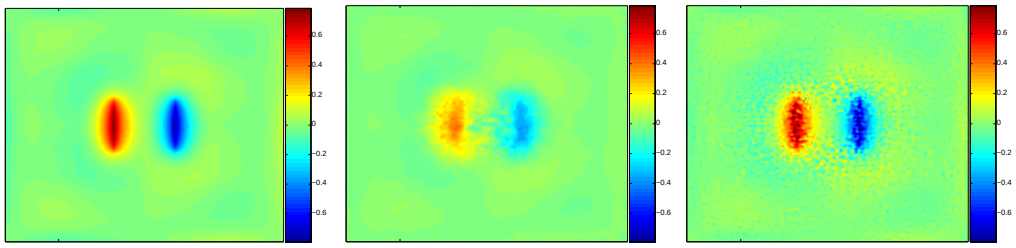

Figure 5.4.: **Left:** Exact curl of velocity field. **Middle:** Estimated curl (Horn&Schunck R+S). **Right:** Estimated curl (2nd-order reg.).

can be expressed as functions of the *stream potential* ϕ and the *velocity potential* ψ:

$$u_{ir} = \nabla\phi = (\frac{\partial\phi}{\partial x}, \frac{\partial\phi}{\partial y})^\top,$$
$$u_{so} = \nabla\psi^\perp = (-\frac{\partial\psi}{\partial y}, \frac{\partial\psi}{\partial x})^\top.$$

In many relevant experimental scenarios one is in fact interested in the functionals ϕ and ψ, and only secondarily interested in the velocity field u. Therefore, it makes sense to rewrite the data term in terms of ϕ and ψ [KMS03][3]

$$E(\phi,\psi) = \int_\Omega (I_2(x + \nabla\phi(x) + \nabla\psi^\perp(x)) - I_1(x))^2 dx \tag{5.7}$$

and separately linearize it for $\nabla\psi^\perp$ and $\nabla\phi$. In [KMS03], auxiliary variables are introduced for the higher-order regularizer

$$E(\phi,\psi,\xi,\omega) = \int_\Omega \lambda(|\text{div } \nabla\phi - \xi|^2 + |\text{curl } \nabla\psi^\perp - \omega|^2) + \mu(||\nabla\xi||^2 + ||\nabla\omega||^2)dx, \tag{5.8}$$

analogous to (5.6). The sum of energy functionals (5.7) and (5.8) is minimized alternatively with respect to ϕ, ψ, ξ and ω. It is shown that the *direct* estimation of the potential functions (by minimizing (5.8)) is superior to their *indirect* estimation (as proposed in [CMP02]) by integration along the stream lines.

Discretization of the Euler-Lagrange equations of (5.8) yields biharmonic operators. It is clear that discretization and numerical solver have to be harmonized in order to be able to prove convergence. [YRMS05] give a sound mathematical background to the minimization of

$$E(\phi,\psi) = \int_\Omega (I_2(x + \nabla\phi(x) + \nabla\psi^\perp(x)) - I_1(x))^2 + \lambda(||\nabla\text{div } \nabla\phi||^2 + ||\nabla\text{curl } \nabla\psi^\perp||^2)$$

by applying mimetic finite differences [HS99, HS97b, HS97a]. The corresponding iterative minimization is provably convergent (subsequent subspace corrections). The discretization allows the encoding of the velocity's boundary values ($u_{\partial\Omega} = \partial_n\psi$); in contrast to the approach presented in [KMS03], the boundary values therefore do not have to be precomputed and subtracted.

[3] The authors subtract the velocity field's laminar component in a *pre*-processing step. Therefore, the Helmholtz decomposition (5.1) holds.

The authors also note that – while every admissible flow field can be uniquely decomposed into its divergent and rotational parts and its boundary values – the traditional higher-order regularization term (5.5) only enforces smoothness of divergence and curl, but does not *directly* enforce smoothness of the boundary values. Therefore they add the penalizer

$$\int_{\partial\Omega} (\partial_n u)^2 dl,$$

which can be expressed in terms of ϕ.

5.1.4. Experimental Evaluation

The aim of this section is to compare first-order regularization and second-order regularization. Section 5.1.2 has already shown that higher-order techniques tend to give better results in real-world fluid flows, as they are much better capable of extracting regions with large divergence or curl. However, figure 5.4 indicates that while the curl is extracted much better with higher-order regularization, the extracted curl is rather noisy, and artifacts are introduced.

First, we describe the data sets used for comparison, then the alternative approaches (and the corresponding parameter settings) are introduced. Finally, we discuss the results.

Data

The experimental evaluation was carried out on the basis of the following two data sets:

- **VSJ:** As in sec. 4.1.4, we will use the eight standard image pairs provided by the Visualization Society of Japan. The image parameters can be taken from table 4.1.
- **Cemagref Synthetic Highly Non-Rigid Image Pair:** This highly non-rigid synthetic PIV image pair was provided by [CH]. The underlying velocity field was computed by a so-called pseudo-spectral code that solves the vorticity transport equation in Fourier space and evaluates a sub-grid model for simulating small-scale turbulent effects on the larger scales of the flow. The image size is 256×256 pixels. The maximum displacement is approximately 3.5 pixels.

Approaches and Parameter Settings

We compared the following first-order and second-order approaches:

- **First-Order Regularization:** We use five resolution levels and nine scale space levels on every resolution level. We set the smoothness parameter λ to $5 \cdot 10^{-3}$, but use no additional brightness fitting term (for H&S R+S). For H&S R+S+B, we added this term and set $\mu = 10$.
- **Higher-Order Regularization:** The VSJ image pairs were analyzed in [CHA+05] using a variational framework combining higher-order regularization and either the optical flow constraint or the integrated continuity equation. For analyzing the Cemagref image pair, we used the higher-order div-curl approach introduced in [YRMS05] (parameters: $\lambda_1 = 0.5, \lambda_2 = 0.05$). Both implementations contain a coarse-to-fine framework with iterated registration.

Results and Discussion

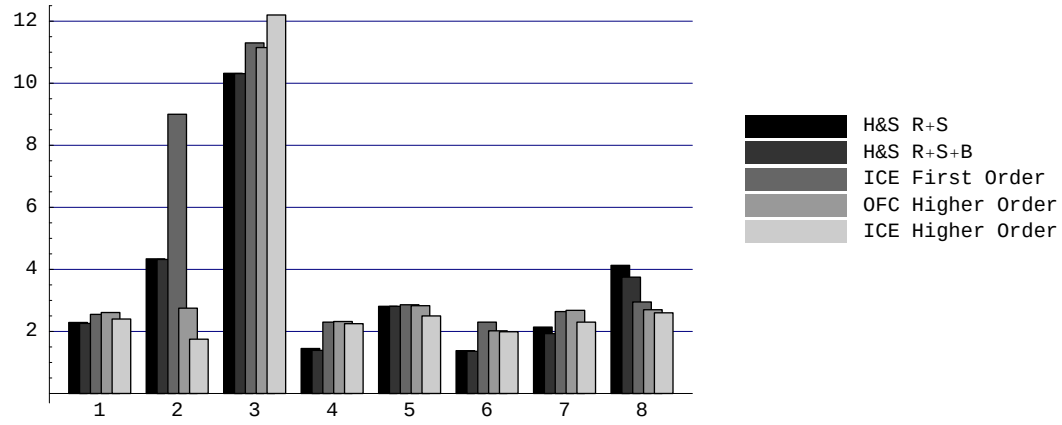

Figure 5.5.: Average relative L_1 error of the different optical flow approaches for the VSJ standard image pairs 01-08.

VSJ Standard Images Let us examine features of both first-order and higher-order regularization by analyzing first the "VSJ standard images" (cf. sec. 4.1.4). Look at fig. 5.5: The error measures for image pairs 02 and 03 are most interesting since they clearly exhibit the different strengths of the approaches:

- The two higher-order regularization approaches give extremely good results for image pair 02. First-order regularization seems to be inappropriate in this case, as it yields too smooth velocity estimates in examples with such a high motion range as present in this image pair.

- For image pair 03 (that has a very low motion range), in turn, first-order regularization seems to be the better choice. Second-order div-curl regularization seems to introduce more artifacts than first-order methods. We will come back to this point in section 5.4, where we will introduce a physically motivated regularization term that can be considered a combination of first-order and second-order regularization.

Cemagref Image Pair Figure 5.6 shows the true velocity field and the vorticity distribution of the Cemagref image pair. Note that the velocity field is highly non-rigid. Figure 5.7 compares the results achieved with first-order and with higher-order regularization. It can be seen clearly that the Horn&Schunck approach has problems at regions where the vorticity is large. This had to be expected, as first-order regularization penalizes high vorticities.

In contrast, the main problems of the second-order approach are at locations where the vorticity is not smooth but changing abruptly. These problems had to be expected, as vorticity variations are penalized by higher-order regularization terms.

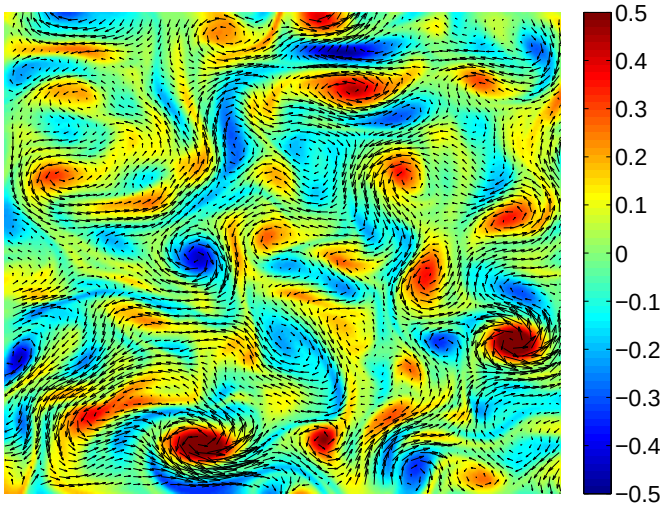

Figure 5.6.: Cemagref Image Pair: Target Velocity and Vorticity

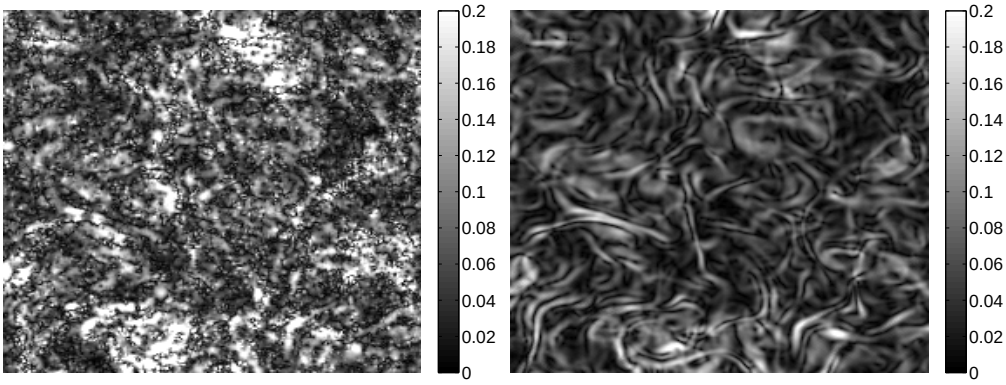

Figure 5.7.: Cemagref Image Pair: Error Measures. **Left:** Horn&Schunck R+S, av. err. = 0.0821 px. **Right:** 2nd-Order Regularization, av. err. = 0.0525 px.

For further experimental results and comparisons with other techniques we refer to sections 5.3.3 and 5.4.3.

Note that we have motivated higher order regularization from the simple observation that there are often compact areas within the flow where divergence or curl are large. Experimental evaluations of turbulent flows confirmed that second-order div-curl approaches are capable of yielding better results than simple first-order methods.

In the next section, we discuss possible physical interpretations of the standard first-order and higher-order regularizers, and will realize that they lack a *sound* physical interpretation. Consequently, we will introduce regularizers with a *direct* physical basis in sections 5.3 and 5.4.

5.2. Discussion: Physical Interpretation of Standard Regularization Terms

In this section, we will analyze standard regularization terms from a *physical* point-of-view. We will first analyze the simple first-order regularization term pioneered by Horn&Schunck (cf. sec. 3.4) and then turn our attention towards higher-order regularization (cf. sec. 5.1). For a short introduction on the governing equations of fluid motion, we refer to appendix C.

First Order Regularization Let us examine the temporal derivative of the kinetic energy of an incompressible fluid. Using (C.14) and assuming that no external forces are acting on the fluid (i.e. $f = 0$), we compute

$$\frac{d}{dt}E_{kin} = \frac{d}{dt}\frac{1}{2}\int_\Omega \rho||u||^2 dx = \int_\Omega \rho u \cdot \frac{Du}{Dt}dx = \int_\Omega (-u\cdot\nabla p + \frac{1}{Re}u\cdot\triangle u)dx \ .$$

Because u is orthogonal to ∇p, we get

$$\frac{d}{dt}E_{kin} = \frac{1}{Re}\int_\Omega u\cdot\triangle u dx = -\frac{1}{Re}\int_\Omega ||\nabla u||^2 dx \ .$$

Therefore, the Horn&Schunck regularization term is just the assumption of a constant kinetic energy over time.

One can also describe the Horn&Schunck regularization term as a special case of the linearized Navier-Stokes equations: Consider the compressible Navier-Stokes equation (C.7). Dropping the nonlinear and the pressure term, considering the stationary case (i.e. $\frac{\partial u}{\partial t} = 0$) and assuming that no forces act on the fluid, yields the so-called Navier-Lamé equation

$$\mu\Delta u + (\lambda+\mu)\nabla(\nabla\cdot u) = 0 \ . \tag{5.9}$$

Let us consider the special case[4], where $\mu > 0$ and $\lambda = -\mu$. (5.9) can be simplified to

$$\mu\Delta u = 0 \ .$$

[4]One can show, however, that this special case is physically *not* plausible: General principles of thermodynamics state that the inequations $\mu \geq 0$, and $\lambda \geq -\frac{2}{3}\mu$ must be valid. For details, we refer to [CC70].

It is clear that the corresponding energy functional minimization is

$$\inf_{u \in V} \int_\Omega \frac{1}{2} \mu |\nabla u|^2 dx \ ,$$

which is just the well-known Horn&Schunck regularization term.
Note that there are three basic drawbacks of this regularizer:

- We have dropped the non-linear terms of the Navier-Stokes equations. These terms are dominant in high Reynolds number environments. We can therefore expect that our regularization term will yield rather high errors in these environments.
- The choice of $\lambda = -\mu$ cannot be physically motivated.
- The body-force is assumed to be zero. Therefore, we just minimize the error between the apparent flow field and our model (that does not include forces).

Higher Order Regularization Letting the body-force vanish does not really make sense - in fact, one should rather be interested in the body force: We should try to find the smallest (in an L2 sense) body force that *explains* the apparent motion.

This can be written as a constrained optimization problem:

$$\begin{aligned} \inf_f & \int_\Omega |f|^2 dx \\ & s.t. \ \ \mu \Delta u = f \ , \end{aligned}$$

which can be easily rewritten as

$$\inf_u \int_\Omega |\mu \Delta u|^2 dx \ . \tag{5.10}$$

Simple calculations show that (5.10) is just a special case of the second-order regularization term (5.5), with $\alpha = \beta := \mu$, and $u = 0$ at the boundaries.

To summarize, both traditional first-order and second-order regularization can in principle be explained by formulas from fluid mechanics. However, neither term is able to describe convection nor do the terms model pressure. Furthermore, the parameter choice $\lambda = -\mu$ is not valid from a physical point of view. In the next section, we will introduce a regularizer that is physically much more plausible: The Stokes equation is the linearized version of the incompressible Navier-Stokes equations. In section 5.4, we finally use the full (incompressible) Navier-Stokes equations (in the form of the vorticity transport equation) for physically consistent regularization.

5.3. Optical Stokes Flow: An Imaging-Based Control Approach

Let us now study a novel optical flow-based approach to *Particle Image Velocimetry (PIV)* that incorporates physical prior knowledge in a more precise and explicit way: All admissible flows for estimation have to satisfy the Stokes equation. In order to estimate the specific flow of apparent velocities of particles in an image sequence, control variables are included and determined

by minimizing a suitable objective function, which relates the flow and the control variables to given image sequence data. We show that our approach not only estimates the flow from a given PIV image sequence, but it estimates pressure and forces acting on the real fluid as well, provided the real flow satisfies the Stokes equation, too.

Our approach draws on the general literature on the control of distributed parameter systems [Lio71]. For specific approaches in connection with fluid dynamics, we refer to [Gun02]. The application of flow-control techniques to image-motion estimation, as presented in this work, is, however, novel.

Concerning the incorporation of physical constraints for flow estimation through image processing, we refer to the short summary given in section 2.2.1 (vi). We have seen that these approaches generally use general-purpose methods for optimization (including simulating annealing, and evolutionary and genetic programming). This indicates, that only little insight into the structure of the problem has been gained (existence, multiplicity and stability of solutions, and related dedicated algorithms). This sharply contrasts with our approach developed below.

The reader may ask: Why we do confine ourselves to Stokes flows, as opposed to flows governed by the full Navier-Stokes equation? In this connection, we wish to point out that we consider, for the first time to our knowledge, a quite difficult *inverse problem* – the joint estimation of a flow along with related physical quantities. This problem is intricate through the *interaction* of various components, although each of them individually behaves in a mathematically simple way. Therefore, to study the computational feasibility and robustness, we have chosen Stokes flows as a first step. Notwithstanding this restriction, our numerical results turned out to be competitive with respect to alternative approaches of current research.

In section 5.3.1 we will present the constrained minimization problem that is being solved - along with the control approach that is used for optimization. We will illustrate the finite-element discretization, the applied numerics, and features of a coarse-to-fine implementation in section 5.3.2. Numerical experiments on ground-truth image pairs as well as on real-world image sequences will be presented in section 5.3.3. We conclude in section 5.3.4 by indicating extensions within the variational control framework.

5.3.1. Approach

Constrained Variational Optical Flow Estimation

Let us again start with the optical flow constraint

$$\int_\Omega \left[\nabla I \cdot u + \partial_t I\right]^2 dx \ . \tag{5.11}$$

Problem (5.11) is not well-posed because *any* vector field with components $\nabla I \cdot u = -\partial_t I, \forall x$, is a minimizer. We have seen in sec. 3.4, that the standard approach is to add a variational term enforcing smoothness of the flow [HS81]

$$\int_\Omega \Big\{(\nabla I \cdot u + \partial_t I)^2 + \alpha\big(|\nabla u_1|^2 + |\nabla u_2|^2\big)\Big\} dx \tag{5.12}$$

or (as we have seen in sec. 5.1.2) smoothness of its divergence and vorticity [Sut94a]

$$\int_\Omega \Big\{(\nabla I \cdot u + \partial_t I)^2 + \alpha\big|\nabla(\nabla \cdot u)\big|^2 + \beta\big|\nabla(\nabla \times u)\big|^2\Big\} dx \ . \tag{5.13}$$

In this chapter, we investigate an alternative method.

Rather than penalizing the estimated flow with a smoothness term, we require as an additional constraint that (5.11) should be minimized, subject to the time-independent[5] Stokes system describing the steady motion of viscous media:

$$\begin{cases} -\mu\Delta u + \nabla p &= f \quad \text{in } \Omega, \\ \nabla \cdot u &= 0 \quad \text{in } \Omega, \\ u &= g \quad \text{on } \Gamma. \end{cases} \tag{5.14}$$

Here, p denotes the pressure, μ the dynamic viscosity, f the body force acting on the fluid, and g the boundary values that are defined on Γ, which denotes the boundary of Ω.

Our objective is to determine a body force f and boundary values g that yield a velocity field u which matches the apparent motion (measured by (5.11)) as well as possible. Note, however, that the minimization of (5.11) subject to (5.14) only enforces vanishing divergence of the flow u. The diffusion term in (5.14) has no impact because f and g can be chosen so that *every* divergence-free velocity field satisfies the Stokes equation. Therefore, we additionally regularize f and g, rendering the whole system mathematically well-posed. As a result, we finally obtain the objective functional

$$J(u,p,f,g) = \int_\Omega \frac{1}{2}\left[\nabla I \cdot u + \partial_t I\right]^2 dx + \int_{\Omega\setminus\Omega_0} \frac{\alpha}{2}|f|^2 dx + \int_\Gamma \frac{\gamma}{2}|\nabla_\Gamma g|^2 d\Gamma \ , \tag{5.15}$$

which is to be minimized subject to (5.14). Ω_0 in the second term in (5.15) denotes regions in the image where we expect large forces to act on the fluid (e.g. interfaces with solids). Therefore, we exclude body force penalization at these locations. $\nabla_\Gamma g$ denotes the component-wise directional derivative of g tangential to the boundary (for the $x_1 = 0$ boundary, e.g., $\nabla_\Gamma g = (\partial g_1/\partial x_2, \partial g_2/\partial x_2)^\top$).

In terms of control theory (e.g. [Gun02]), the approach can be summarized as follows: We wish to find an optimal state (u,p) and optimal distributed controls (f,g), so that functional J (5.15) is minimized subject to u, p, f, and g satisfying the Stokes system (5.14).

Optimality Conditions

To derive the *optimality system* for determining optimal solutions to (5.15), (5.14), we transform the constrained optimization problem into an unconstrained optimization with the Lagrangian function

$$L(u,p,f,g,w,r,\xi) = J(u,p,f,g) \tag{5.16a}$$

$$- \int_\Omega w^\top(-\mu\Delta u + \nabla p - f) + r(\nabla \cdot u)dx \tag{5.16b}$$

$$- \int_\Gamma \xi^\top(u-g)d\Gamma \tag{5.16c}$$

[5] Note that we confine ourselves to the time-independent case as we want to analyze image pairs only, and therefore have no additional information about the temporal evolution of the velocity.

and corresponding multipliers w, r, ξ. The first-order necessary conditions then yield the optimality system, which determines optimal states and controls:

$$\begin{cases} -\mu\Delta u + \nabla p &= f \quad \text{in } \Omega, \\ \nabla \cdot u &= 0 \quad \text{in } \Omega, \\ u &= g \quad \text{on } \Gamma \end{cases} \tag{5.17a}$$

$$\begin{cases} \mu\Delta w + \nabla r &= -(\nabla I^\top u + \partial_t I)\nabla I \quad \text{in } \Omega, \\ \nabla \cdot w &= 0 \quad \text{in } \Omega, \\ w &= 0 \quad \text{on } \Gamma \end{cases} \tag{5.17b}$$

$$\begin{cases} w + \alpha f &= 0 \quad \text{in } \Omega\backslash\Omega_0, \\ w &= 0 \quad \text{in } \Omega_0, \\ rn - \mu\frac{\partial w}{\partial n} - \gamma\Delta_\Gamma g &= 0 \quad \text{on } \Gamma\,, \end{cases} \tag{5.17c}$$

where $\Delta_\Gamma g$ is the 1D Laplacian of g, tangential to the boundary (for the $x_1 = 0$ boundary, e.g., $\Delta_\Gamma g = (\partial^2 g_1/\partial x_2^2, \partial^2 g_2/\partial x_2^2)^\top$). The *state equation* (5.17a) results from taking the Gâteaux derivative of (5.16) in the direction of the Lagrange multipliers, reproducing the Stokes equation (5.14). Equation (5.17b) is the *adjoint equation*. It specifies the first-order necessary conditions with respect to the state variables u and p. Note that this equation has the same structure as (5.17a) with just the variables replaced by the adjoint velocity w and the adjoint pressure r. Consequently, we can use the same numerical algorithm to solve (5.17a) and (5.17b). The third system of equations (5.17c) states the *optimality condition*, which is the necessary condition for the gradient of the objective functional with respect to the controls to vanish at the optimum. Next, we state the optimization problem for solving (5.17). Discretization and numerical solution of subproblems (5.17a) and (5.17b) are detailed in section 5.3.2.

Optimization Algorithm

Due to the large number of unknowns in the optimality system (5.17), we decouple the state system (5.17a) and the adjoint system (5.17b), and apply the *gradient method* for computing the solution of the optimal control problem. Let us therefore first define the gradients of our objective functional with respect to the body force f and with respect to the boundary values g:

When we change the distributed control f to $f + \epsilon\tilde{f}$, where $\tilde{f}$ is arbitrary, this change in f induces the state to change from $(u,\, p)$ to $(u + \epsilon u_f,\, p + \epsilon p_f)$. The change u_f in the state is determined by the state system, i.e. we have that

$$\begin{cases} -\mu\triangle(u + \epsilon u_f) + \nabla(p + \epsilon p_f) &= f + \epsilon\tilde{f} \\ \nabla \cdot (u + \epsilon u_f) &= 0 \\ (u + \epsilon u_f) &= g \qquad on\ \Gamma\,. \end{cases} \tag{5.18}$$

For $\epsilon \to 0$ this leads to the so-called sensitivity equation

$$\begin{cases} -\mu\triangle u_f + \nabla p_f &= \tilde{f} \\ \nabla \cdot u_f &= 0 \\ u_f &= 0 \quad on\ \Gamma\,. \end{cases} \tag{5.19}$$

This equation says that an infinitesimal change of the control function f in the "direction" of $\tilde{f}$ induces the infinitesimal change in the "directions" of u_f and p_f. Next, let us derive a formula

for the change in the functional $J(u, f)$ of (5.15), effected by an infinitesimal change in the direction $\tilde{f}$ in the control f. We will keep track of both the explicit dependence of J on f and the implicit dependence through the state (u, p):

$$\langle \partial_f J, \tilde{f} \rangle = \frac{\partial J(u + \epsilon u_f, p + \epsilon p_f, f + \epsilon \tilde{f})}{\partial \epsilon}\Big|_{\epsilon=0} = \int_\Omega \Big\{ (\nabla I^\top u + I_t)\nabla I^\top u_f + \alpha f^\top \tilde{f} \Big\} dx \ .$$

Substitution of the first adjoint equation (5.17b) yields

$$\langle \partial_f J, \tilde{f} \rangle = \int_\Omega \Big\{ -(\mu \triangle w + \nabla r)^\top u_f + \alpha f^\top \tilde{f} \Big\} dx.$$

Integrating by parts yields

$$\langle \partial_f J, \tilde{f} \rangle = \int_\Omega \Big\{ -\mu \triangle u_f^\top w + \nabla \cdot u_f r + \alpha f^\top \tilde{f} dx + \int_\Gamma \mu(-u_f^\top \frac{\partial w}{\partial n} + w^\top \frac{\partial u_f}{\partial n}) + u_f^\top r n \Big\} d\Gamma. \tag{5.20}$$

The boundary integral is zero, as w and u_f are zero at the boundaries. Now we can substitute the sensitivity equation (5.19) into (5.20)

$$\langle \partial_f J, \tilde{f} \rangle = \int_\Omega \Big\{ (\tilde{f} - \nabla p_f)^\top w + \alpha f^\top \tilde{f} \Big\} dx$$

Again, integration by parts and substitution of the second adjoint equation finally leads to

$$\langle \partial_f J, \tilde{f} \rangle = \int_\Omega \Big\{ w^\top \tilde{f} + \alpha f^\top \tilde{f} \Big\} dx \ . \tag{5.21}$$

An analogous derivation yields the gradient of J in g-direction:

$$\langle \partial_g J, \tilde{g} \rangle = \int_\Gamma \Big\{ -\mu \Big(\frac{\partial w}{\partial n} \Big)^\top \tilde{g} + r n^\top \tilde{g} + \gamma \nabla_\Gamma g^\top \nabla_\Gamma \tilde{g} \Big\} d\Gamma \ . \tag{5.22}$$

Now that we have found formulas for the gradients of our objective functional, we can develop the gradient algorithm:

We start with a velocity field $u = \mathbf{0}$ (or with any other initial value) and solve the adjoint equation. The gradient of the functional with respect to g and f, respectively (cf. (5.21) and (5.22)), is

$$\partial_f J = w + \alpha f \ , \tag{5.23a}$$

$$\partial_g J = rn - \mu \frac{\partial w}{\partial n} - \gamma \Delta_\Gamma g - \frac{n}{|\Gamma|} \int_\Gamma (-\mu \frac{\partial w}{\partial n} + rn - \gamma \Delta_\Gamma g) \cdot n d\Gamma; \ . \tag{5.23b}$$

Note that r is determined by the adjoint equation (5.17b) only up to a constant. (5.23b) chooses this constant so that the update of g satisfies the compatibility condition $\int_\Gamma g \cdot n d\Gamma = 0$ (sum of inflow = sum of outflow – must be valid for incompressible fluids). Having updated the controls, we solve the state equation and proceed to the next iteration. After convergence of the algorithm, (5.17c) is satisfied, too.

Experiments have shown that using two individual and adaptive step sizes for f and g, respectively, is computationally both more reliable and efficient. This leads to the algorithm 1 listed below. Note that the step-size parameters τ_f and τ_g are automatically selected by the algorithm.

Algorithm 1 Gradient Algorithm

set initial $u = 0$
choose tolerance ϵ
$\tau_f := 1$, $\tau_g := 1$, $f_0 := 0$, $g_0 := 0$
repeat
  solve (5.17b) for (w,r)
  $f_i = f_{i-1} - \tau_f(\alpha f_{i-1} + w)$
  solve (5.17a) for u
  if $J(u, f_i) \leq J(u, f_{i-1})$ **then**
    τ_f:=τ_f/2
    GOTO 6
  else
    τ_f:=3/2τ_f
  end if
  solve (5.17b) for (w,r)
  $g_i = g_{i-1} - \tau_g\big[rn - \mu\frac{\partial w}{\partial n} - \gamma\Delta_\Gamma g - \frac{n}{|\Gamma|}\int_\Gamma(-\mu\frac{\partial w}{\partial n} + rn - \gamma\Delta_\Gamma g)^\top n d\Gamma\big]$
  solve (5.17a) for u
  if $J(u, g_i) \leq J(u, g_{i-1})$ **then**
    τ_g:=τ_g/2
    GOTO 15
  else
    τ_g:=3/2τ_g
  end if
until $|J(u, f_i, g_i) - J(u, f_{i-1}, g_{i-1})|/|J(u, f_i, g_i)| < \epsilon$

Relaxing the Assumption of a Vanishing Divergence

Due to out-of-plane motion (that can hardly be totally avoided), the assumption of a vanishing divergence will often not hold in practice. Optical Stokes Flow's strict enforcement of a vanishing divergence will lead to large errors in the velocity field if the incompressibility assumption is incorrect.

Let us therefore modify the prior knowledge that we use:

$$\begin{cases} -\mu\Delta u + \nabla p &= f & \text{in } \Omega, \\ \nabla \cdot u &= -\frac{\partial u_3}{\partial x_3} & \text{in } \Omega, \\ u &= g & \text{on } \Gamma\,, \end{cases} \tag{5.24}$$

where u_3 is the out-of-plane component (component in x_3 direction). Note, however, that u_3 is unknown and cannot be extracted from the algorithm's 2D input data. This is why we need an additional assumption: $d = \frac{\partial u_3}{\partial x_3}$ should be small. This leads to the overall optimization problem:

$$J(u,p,f,g,d) = \int_\Omega \frac{1}{2}\big[\nabla I \cdot u + \partial_t I\big]^2 dx + \int_{\Omega\setminus\Omega_0} \frac{\alpha}{2}|f|^2 dx + \int_\Omega \frac{\beta}{2}|d|^2 dx \int_\Gamma \frac{\gamma}{2}|\nabla_\Gamma g|^2 d\Gamma\,,$$

which is to be minimized subject to (5.24). The corresponding optimization algorithm is equivalent to the one presented in sec. 5.3.1. The gradient of the additional control d completes (5.23)

$$\partial_d J = \beta d + r.$$

5.3.2. Discretization and Implementation

Solving the Subproblems

In order to apply algorithm 1 to the optimality system (5.17), we have to solve two saddle point problems corresponding to the state equation (5.17a) and to the adjoint equation (5.17b), respectively. In this section, we explain how these problems are discretized and numerically solved.

The unique vector field $u(x_1, x_2)$ solving (5.17a) is determined by the variational system

$$\begin{aligned} a(u,\tilde{u}) + b(p,\tilde{u}) &= (f,\tilde{u})\,, \quad \forall \tilde{u} \\ b(\tilde{p},u) &= 0\,, \quad \forall \tilde{p} \end{aligned} \tag{5.25}$$

and a similar variational system determines the unique solution w to (5.17b). Accordingly, we define for the Stokes problem and for the adjoint problem, respectively, bilinear forms and linear forms:

$$a_{St}(u,\tilde{u}) := \int_\Omega \mu\nabla u \cdot \nabla\tilde{u} dx \qquad a_{Adj}(w,\tilde{w}) := \int_\Omega -\mu\nabla w \cdot \nabla\tilde{w} dx \tag{5.26}$$

$$b_{St}(p,\tilde{u}) := -\int_\Omega p\nabla\cdot\tilde{u} dx \qquad b_{Adj}(r,\tilde{w}) := -\int_\Omega r\nabla\cdot\tilde{w} dx \tag{5.27}$$

and the right hand sides:

$$(f_{St},\tilde{u}) := \int_\Omega f\cdot\tilde{u} dx \qquad (f_{Adj},\tilde{w}) := \int_\Omega -(\nabla I^\top u + \partial_t I)\nabla I\cdot\tilde{w} dx\,. \tag{5.28}$$

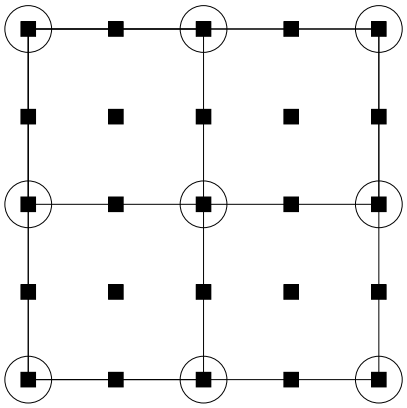

Figure 5.8.: Sketch of 2D Taylor-Hood elements: biquadratic velocity elements (squares) and bilinear pressure elements (circles).

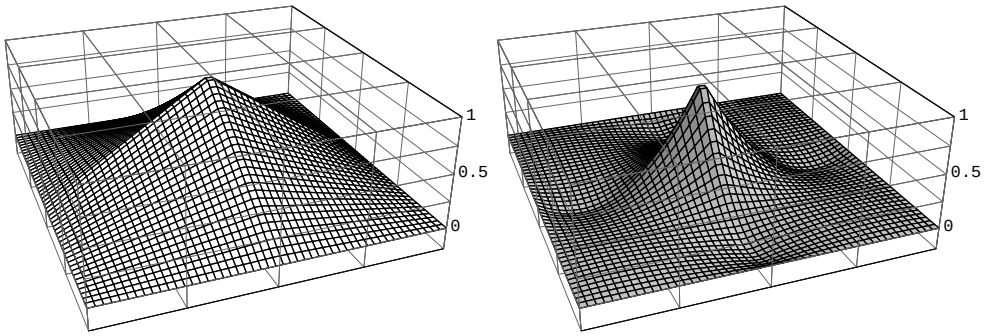

Figure 5.9.: **Left:** Basis function ϕ of a bilinear finite element. **Right:** Basis function ψ of a biquadratic finite element.

We choose a regular tessellation of the image domain Ω and discretize (5.25) using finite elements. It is well-known from computational fluid dynamics that standard first-order finite element discretizations may result in non-physical pressure oscillations or even in so-called locking effects, where the zero velocity field is the only one satisfying the incompressibility condition.

Therefore, when solving the Stokes problem, mixed finite elements are traditionally used. An admissible choice is the so-called Taylor-Hood element based on a square reference element with nine nodes (fig. 5.8). Each component of velocity fields is defined in terms of piecewise quadratic basis functions ψ located at each node, whereas pressure fields are represented by linear basis functions ϕ attached to each corner node. It can be shown that Taylor-Hood elements fulfill the so-called Babuska-Brezzi condition [BF91], that is the discretized problem is well-posed and numerically stable. Appendix B gives more details about saddle point problems and the mixed finite element method.

Indexing each velocity node (squares of fig. 5.8) by $1, 2, ..., N$, we obtain

$$u_1(x_1, x_2) = \sum_{i=1}^{N} u_1^i \psi^i(x_1, x_2)$$

and similarly for $u_2(x_1, x_2)$ and the components of $w, \tilde{u}, \tilde{w}$. By analogy, we obtain for the M pressure nodes (circles of fig. 5.8)

$$p(x_1, x_2) = \sum_{i=1}^{M} p^i \phi^i(x, y)$$

and similarly expressions for $p, r, \tilde{p}, \tilde{r}$. Hence, each function $u, w, \tilde{u}, \tilde{w}$ is represented by $2N$ real variables, and each function $p, r, \tilde{p}, \tilde{r}$ is represented by M real variables. For the sake of simplicity, we will use the same symbols to denote these vectors. (5.25) then reads

$$\begin{aligned} Au \cdot \tilde{u} + B^\top p \cdot \tilde{u} &= f \cdot \tilde{u}\,, \quad \forall \tilde{u} \\ Bu \cdot \tilde{p} &= 0\,, \quad \forall \tilde{p}\,. \end{aligned}$$

Hence, we obtain the discretized Stokes system

$$\begin{aligned} Au + B^\top p &= f \\ Bu &= 0 \end{aligned} \tag{5.29}$$

and a similar system for the adjoint equation. The $2N \times 2N$-Matrix A factorizes into

$$A = \begin{pmatrix} A_{11} & 0 \\ 0 & A_{22} \end{pmatrix} \tag{5.30}$$

where by virtue of (5.26):

$$\begin{aligned} (A_{11})_{k,l} &= a((\psi_k, 0)^\top, (\psi_l, 0)^\top) \\ (A_{22})_{k,l} &= a((0, \psi_k)^\top, (0, \psi_l)^\top)\,. \end{aligned}$$

The $M \times 2N$-Matrix B factorizes into

$$B = \begin{pmatrix} B_1 & B_2 \end{pmatrix}$$

where by virtue of (5.27)

$$\begin{aligned} (B_1)_{k,l} &= b(\phi_k, (\psi_l, 0)^\top) \\ (B_2)_{k,l} &= b(\phi_k, (0, \psi_l)^\top)\,. \end{aligned}$$

Finally, the 2N-vector f factorizes into $f = (f_1^\top, f_2^\top)^\top$ where by virtue of (5.28)

$$\begin{aligned} (f_1)_k &= (\psi_k, 0) \\ (f_2)_k &= (0, \psi_k)\,. \end{aligned}$$

In order to numerically solve the saddle point problem (5.29), we employ the Uzawa algorithm (cf. appendix B.3).

Note that A is just the system matrix of the Poisson equation. For 2D problems, A can be split into two systems (one for every dimension, cf. (5.30)), that can be solved in parallel.

Coarse-to-fine Approach

Due to the Taylor series linearization inherent in the optical flow constraint, only slow motion can be accurately computed by minimizing (5.11) (approx. up to 1 pixel between exposures). This is why we apply our approach to a multi-scale representation of the image data I: We first compute a coarse motion field by using only low spatial frequency components and undo the motion, thus roughly stabilizing the position of the image over time. Then the higher frequency sub-bands are used to estimate the optical flow on the warped sequence. Combining this optical flow correction with the previously computed optical flow yields a refined overall optical flow estimate. This process is repeated at finer spatial scales until the original image resolution is reached. For further details about coarse-to-fine optical flow estimation and symmetric image warping we refer to section 4.1.3.

Let u denote the overall velocity that results from our computations, u_{old} the current estimate of this overall velocity, and $\partial_t I_w$ the temporal derivative computed as difference between the second image - warped with u_{old} - and the first image. Then (5.11) can be reformulated as

$$J(u) = \int_\Omega \frac{1}{2}\big(\nabla I^\top (u - u_{old}) + \partial_t I_w\big)^2 dx \ .$$

Note that motion over the image boundary Γ prevents the computation of the spatial and temporal gradients of the warped image I_w at specific locations. In order to avoid error-prone filling-in heuristics (that use gradient information from surrounding areas), we simply omit the evaluation of the data term at these particular locations.[6] This reformulation does not affect the state system and the optimality condition. The adjoint system, on the other hand, is transformed into

$$\begin{cases} \mu\Delta w + \nabla r & = -\big(\nabla I^\top (u - u_{old}) + \partial_t I_w\big)\nabla I & \text{in } \Omega \\ \nabla \cdot w & = 0 & \text{in } \Omega \\ w & = 0 & \text{on } \Gamma \ . \end{cases}$$

We could now - as we did in algorithm 1 - start at every resolution level with an initial zero velocity field. This is a poor initialization, however: We know that u_{old} is a good approximation of the true velocity field: Therefore, we solve (5.17a) for (u, p) before the first iteration of every resolution level, using bilinearily interpolated versions of f and g from the preceding level. We obtain an initial velocity u that both satisfies (5.17a) and is a good approximation of the true velocity field.

As solving the Optical Stokes Flow problem is computationally rather expensive, an alternative procedure is to use a simpler (and therefore faster) approach on the coarser pyramid levels (e.g. Horn&Schunck) and to solve (5.17) only on the highest pyramid level.

5.3.3. Experimental Evaluation

This experimental section is divided into two main parts:

- In sec. 5.3.3 (i) we present synthetic experiments that fulfill the Stokes equation (i.e. creeping flows). Besides the question of the accuracy of our method, we want to go further into the question of how meaningful the asserted estimates for pressure p and body force f prove to be.

[6] Note that due to the regularizer, we will still get reliable velocity estimates at these locations.

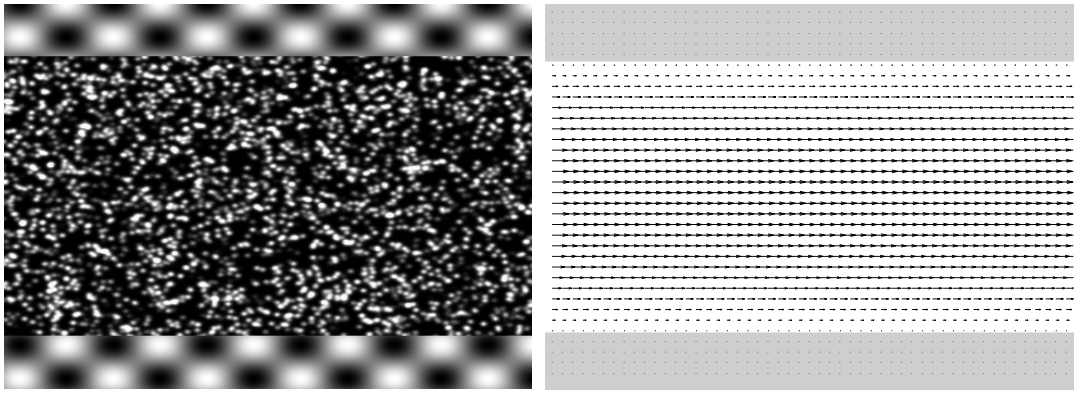

Figure 5.10.: Poiseuille Flow: Incompressible Newtonian fluid with constant density and viscosity flows between two parallel plates. This creeping flow satisfies the Stokes equation. **Left:** Synthetic PIV image. **Right:** Target velocity field.

- In sec. 5.3.3 (ii) we show (for synthetic and real-world image pairs) that one can also achieve good velocity estimates for highly non-rigid flows. In these cases, however, we cannot expect the body-force and the pressure distribution to contain physically relevant information.

(i) Stokes Flows

We have selected two flow scenarios for which analytic solutions exist due to symmetry: we analyze *Poiseuille Flow*, which is a viscous flow between two parallel plates, and section *Flow in an Annular Gap* examines the viscous flow between two infinitely long cylinders.

Poiseuille Flow We consider an incompressible Newtonian fluid with constant density and viscosity that flows between two parallel plates (at $x_2 = 0$ and $x_2 = h$) with infinite width. The x-axis points in the direction of the flow. The velocity distribution for such a system is given by (e.g. [LL52])

$$u(x_2) = -\frac{1}{2\mu}\frac{\partial p}{\partial x_1}\left(\frac{h^2}{4} - (x_2 - \frac{h}{2})^2\right).$$

This means that we can expect a parabolic velocity profile, with the largest velocity in the middle between the two plates. For our synthetic experiment, we chose $\mu = 1$, $h = 257\,px.$, and $\frac{\partial p}{\partial x} = -1 \cdot 10^{-3}$. This choice yields a maximum velocity of 8.256 pixels between two exposures. Figure 5.10 shows the synthetic image, and the target velocity field with which the image was warped in order to get a synthetic image pair. We used the same techniques as described in [ONK00a] (10.000 particles, 3 px. average particle diameter, 1 px. standard deviation). As the Poiseuille flow is truly 2D, the third component is zero everywhere. Please note that while in numerical hydrodynamics one is accustomed to small mesh sizes ($s << 1$) and small volumes ($VOL \approx 1$), we measure in terms of pixels here. This is why the parameter choice may appear uncommon.

In a first experiment, we set $\alpha = 0.001$ and $\gamma = 0.002$, and we penalized the body force everywhere (i.e. $\Omega_0 = \emptyset$). Figure 5.11 shows the reconstructed velocity component u. The

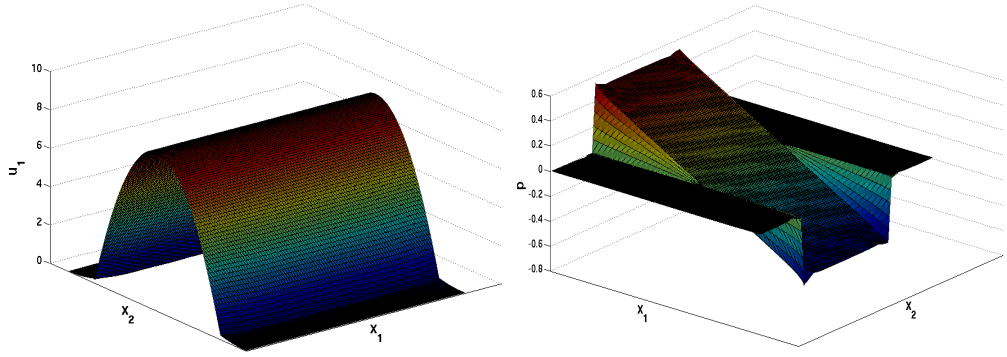

Figure 5.11.: **Left:** Optical Stokes Flow is able to reconstruct the parabolic velocity profile extremely accurately. **Right:** Using fluid-mechanics priors, even the true pressure distribution can be reconstructed. Note the linear pressure decrease in flow direction.

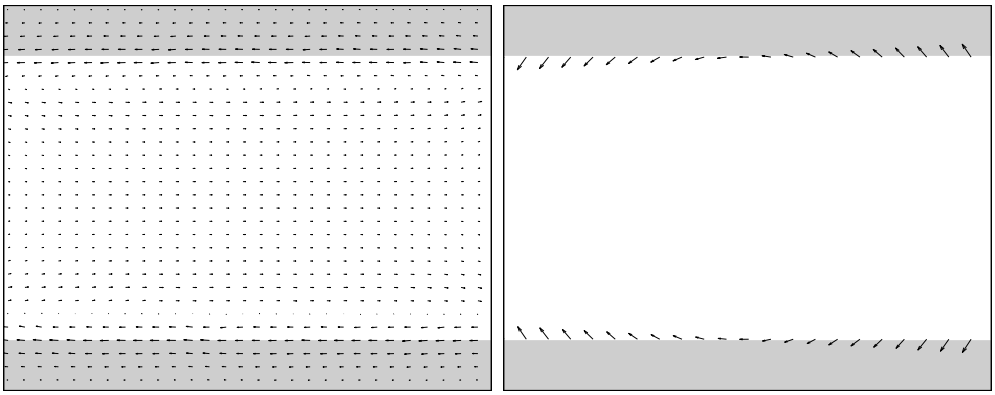

Figure 5.12.: Reconstructed body force. Note that the arrows are scaled in order to be visible. If we specify regions at which forces are expected to act on the fluid (i.e. solid-fluid interfaces), even forces that act on the boundary can be reconstructed. **Left:** f is penalized everywhere (scaling factor: 2000). **Right:** no penalization of f at the interfaces (scaling factor: 50).

estimated velocity is almost exact (cf. also fig. 5.13), its RMS error is $\approx 0.0734px$.. However, fig. 5.13 also shows that there are problems at the boundaries of the plates. They are caused by the forces acting on the fluid. In fact, the locations where forces act on the fluid are just boundaries of the plates: The pressure-induced force acts orthogonally to the interfaces and is

$$f_x = -pn \ . \tag{5.31}$$

The frictional force at the interfaces acts in opposite flow direction and is

$$f_y = \frac{h}{2}\frac{\partial p}{\partial x} \ . \tag{5.32}$$

Figure 5.11 (left) shows that the algorithm has in fact detected a force at the interfaces that acts in opposite flow direction. However, the method also detects a (smaller) force in the middle of the pipe that acts in flow direction. The reason for this error is quite obvious: In (5.15), we added constraints on the body force that penalize the L_2 norm of f. The correct body force, however, has an extremely high L_2 norm at the interfaces. In order to yet compute a reliable body force – and thus also pressure estimates, as the pressure depends on velocity *and* body force – we have to tell the algorithm at which locations forces are likely to act on the fluid. Then we can exclude the body force penalization at these locations.

Accordingly, in a second experiment, we switched off body force penalization at the interfaces of the two parallel plates (at $x_2 = 0$, and $x_2 = h$). The results can be seen in fig. 5.12 (right): The reconstructed body force is reasonable, the share that acts in reverse flow direction is the frictional force (cf. (5.32)) and the part that acts orthogonally to the flow direction is the pressure-induced force (cf. (5.31)).

Figure 5.13 shows that also the RMS error has decreased considerably (RMS $\approx 0.0212\,px$). Note that there are still errors at the ends of the interfaces; the reason for these errors is the regularization of the boundary values g (cf. (5.15)). The smoothness of the boundary values enforced by (5.15) deviates from reality at these locations.

Figure 5.11 shows the reconstructed pressure field on the right. Taking a closer look at the pressure derivative in flow direction (cf. fig. 5.14), we see that the pressure derivative inside the tube is approx. $4 \cdot 10^{-3}$, which is the correct reconstruction. We point out that due to the mixed finite element discretization, the resolution of the pressure field is smaller than the resolution of the reconstructed velocity. Therefore, the pressure derivative has to be scaled with the factor 4.

Flow in an Annular Gap Suppose an incompressible Newtonian fluid flows steadily within the annular gap of two infinitely long cylinders (R_1, R_2). The outer cylinder is fixed, while the inner cylinder rotates with angular speed Ω. The velocity distribution for such a setting is given by (e.g. [LL52])

$$v(r) = \frac{\Omega R_2^2}{R_2^2 - R_1^2} r + \frac{\Omega R_1^2 R_2^2}{R_2^2 - R_1^2} \frac{1}{r} \ .$$

The pressure is constant.[7] For our synthetic example, we have chosen $R_1 = 100$, $R_2 = 220$, $\Omega = 0.1$ and $\mu = 1$. This leads to a maximum displacement of 10 pixels between two exposures. Figure 5.15 shows the synthetic image as well as the target velocity field. We set $\alpha = 0.001$, $\gamma = 0.002$, and $\Omega_0 = \emptyset$ in a first experiment.

[7] This is only true when solving the problem with the Stokes equation. If we had used the Navier-Stokes equations, the pressure distribution would read $\frac{\partial p}{\partial r} = \frac{\sigma v^2}{r}$.

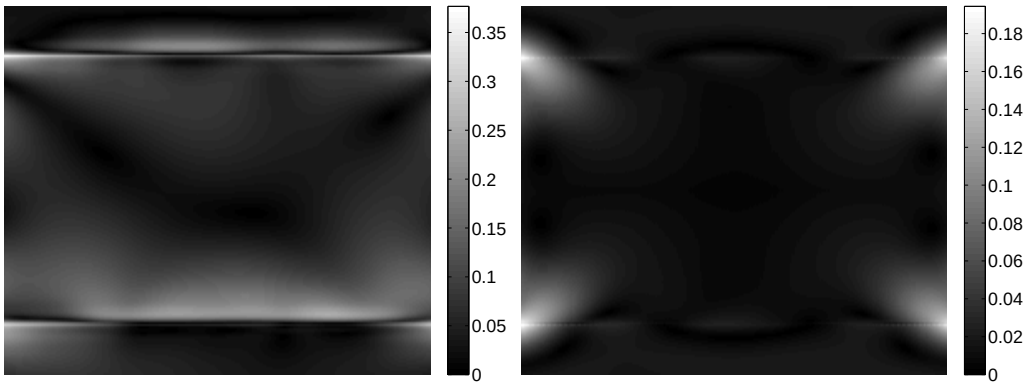

Figure 5.13.: **Left:** If we penalize f everywhere, the mean RMS error is 0.0734 pixels. **Right:** By specifying the solid-liquid interfaces, the RMS error decreases to $0.0212px$.

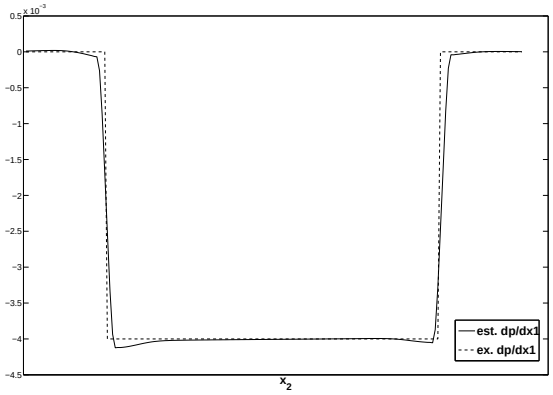

Figure 5.14.: Profile of the averaged pressure derivative in flow direction. Note that the linear pressure decrease in flow direction (-4×10^{-3} inside the pipe) is recovered extremely accurately.

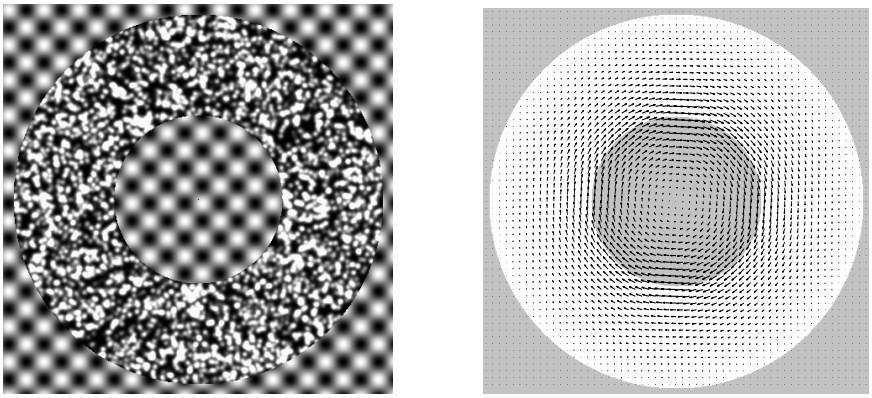

Figure 5.15.: Creeping Flow between two rotating cylinders. **Left:** Synthetic PIV image. **Right:** Target Velocity field.

Figure 5.16 shows the reconstructed velocity field. The velocity field looks very reasonable. Figure 5.20, however, shows clearly a large error at the cylinder interfaces, in particular at the interface of the inner cylinder. This error occurs for the same reason as in the preceding example. The acting force is very large and punctiform at the inner cylinder interface. This contradicts the assumptions made in (5.15).

We next chose the same approach as in the preceding section to deal with the problem, that is we avoided body force penalization at the boundaries. This leads to very good results. The average RMS error decreases to $0.0079px$. (cf. fig. 5.20). Fig. 5.17, right, shows that the velocity profile agrees exactly with the analytically computed profile. The tangential and orthogonal forces at the two cylinder interfaces also correspond very well with the analytically computed forces. The forces in tangential direction read:

$$f_{R_1} = -2\mu \frac{\Omega R_2^2}{R_2^2 - R_1^2} \ ,$$
$$f_{R_2} = -2\mu \frac{\Omega R_1^2}{R_1^2 - R_2^2} \ .$$

The pressure is zero everywhere (cf. fig. 5.17, left). There are only minor problems at the cylinder boundaries due to an increased numerical sensitivity. A reason is that we deliberately omitted regularization of f at these locations, for the sake of accurate reconstruction.

Noise and Robustness To examine the robustness of our reconstruction approach with respect to image noise, we repeated the annular gap experiment (section 5.3.3) but superimposed white noise with a variance of up to 50% of the grayvalue range.

Figure 5.21 shows that the RMS velocity error increases moderately only as a function of noise variance. This result proves a pronounced robustness of our approach. Most remarkable is the observation that the accuracy of the reconstruction appears to be *beyond* what can be extracted from the raw data *without* any physical prior knowledge.

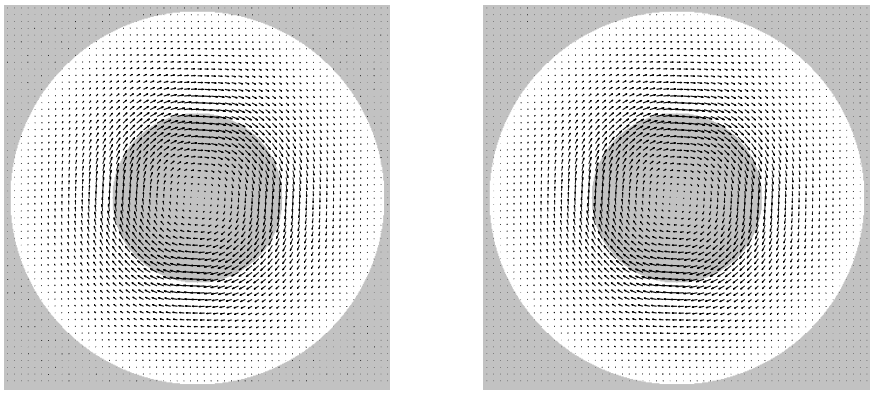

Figure 5.16.: From the visual impression, both velocity fields resemble the true motion extremely well. **Left:** Reconstructed velocity field (penalization everywhere) **Right:** Reconstructed velocity field (no penalization on boundary).

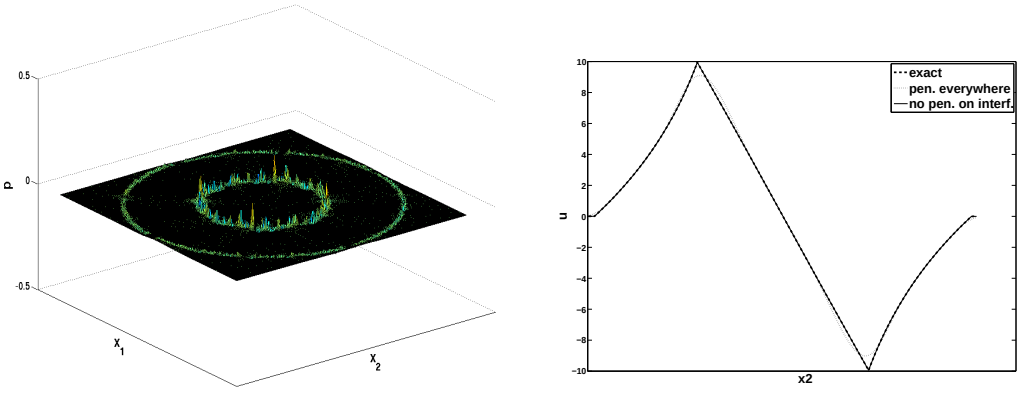

Figure 5.17.: **Left:** Reconstructed pressure distribution (no penalization on boundary): p is zero almost everywhere, some numerical instabilities at the interfaces. **Right:** Profile of the velocity distribution. Note that, if we penalize everywhere, the reconstructed velocity field is much too smooth.

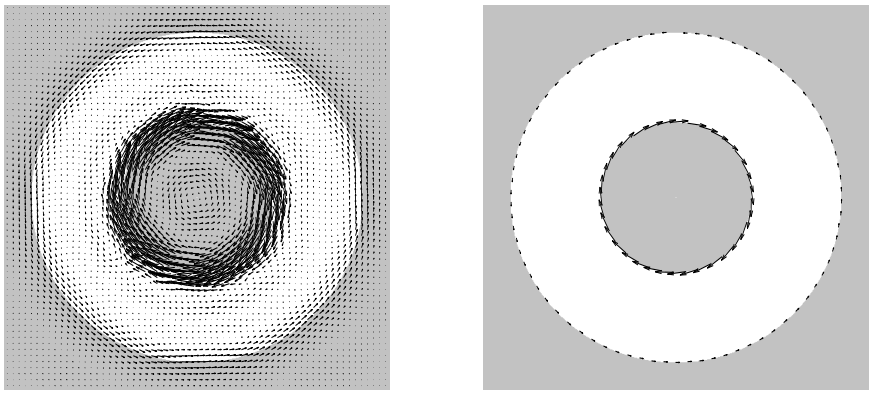

Figure 5.18.: Reconstructed body force. **Left:** Penalization everywhere (force scaled by 10.000). **Right:** No penalization on interfaces (force scaled by 100). The frictional forces at the cylinder interfaces are extracted correctly.

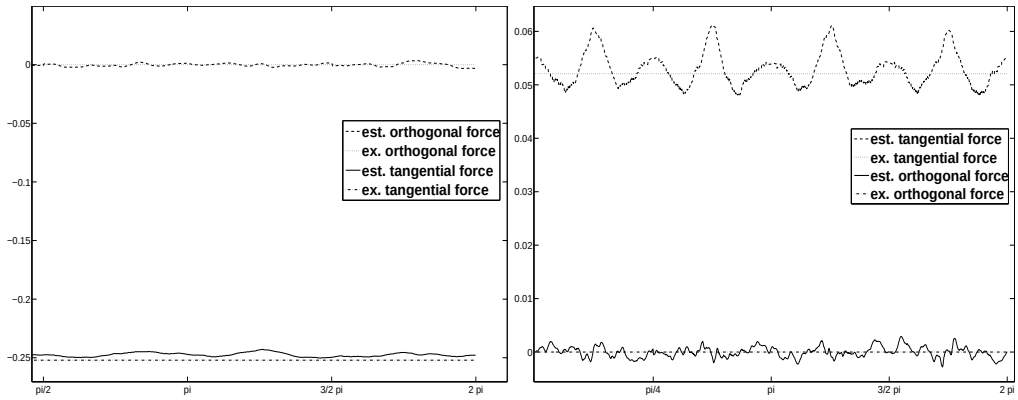

Figure 5.19.: Optical Stokes Flow is able to compute the forces that act on the cylinder interfaces. **Left:** Tangential and orthogonal forces at the inner cylinder boundary. **Right:** Tangential and orthogonal forces at the outer cylinder boundary.

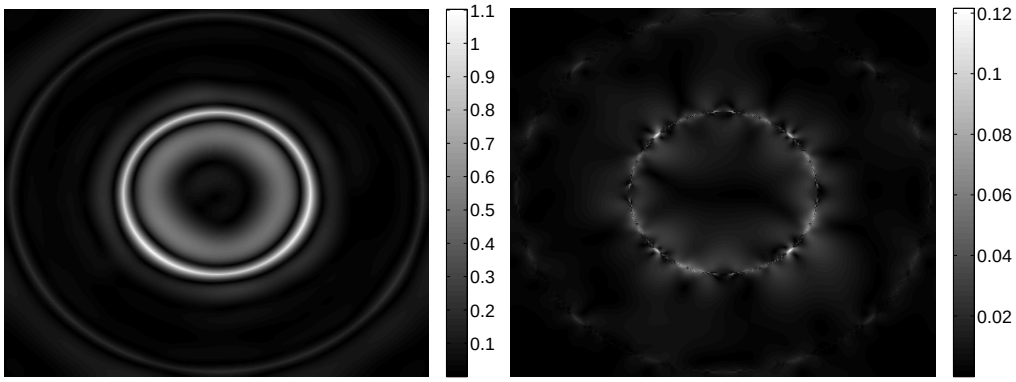

Figure 5.20.: **Left:** RMS velocity error (av. 0.310 px.) when penalizing the body force everywhere. **Right:** RMS velocity error (av. 0.0079 px.) when no penalization on the cylinder boundaries is performed.

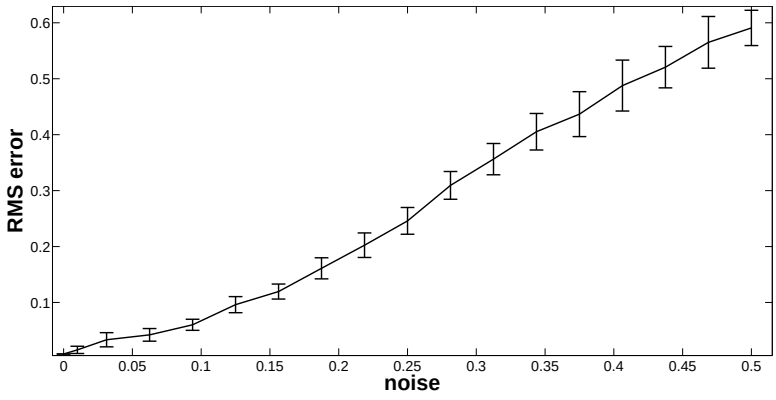

Figure 5.21.: RMS velocity error (in pixels) when adding different amounts of Gaussian noise (zero mean and variance up to 50%).

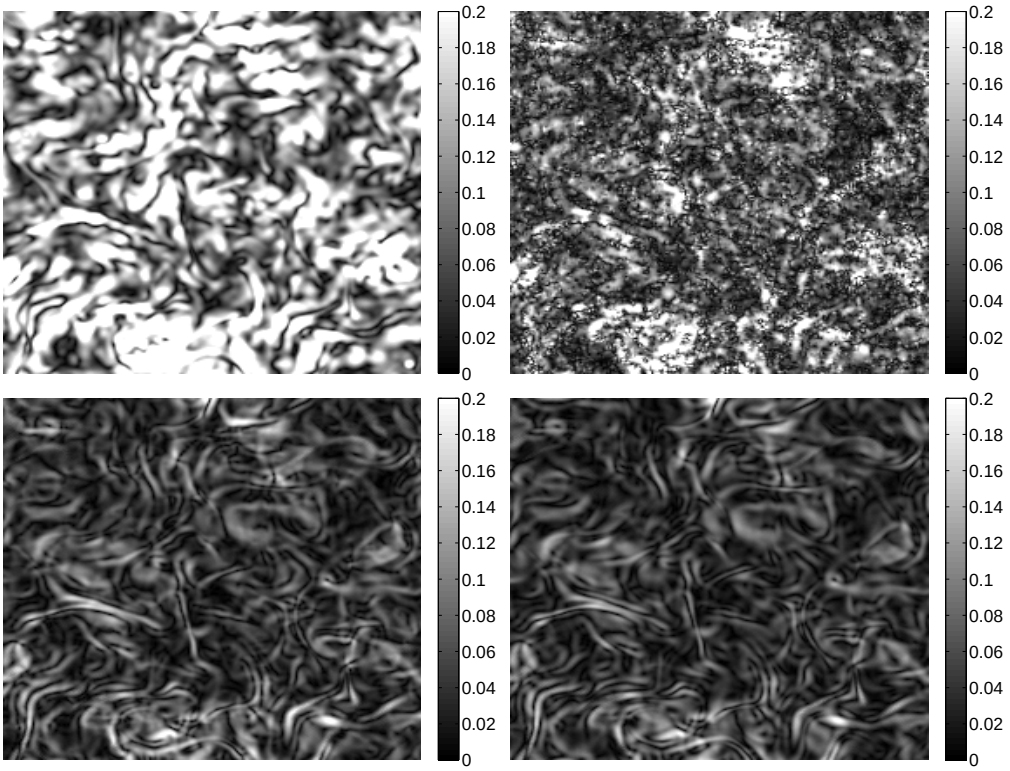

Figure 5.22.: **Top Left:** Multi-Pass Cross-Correlation (DaVis), av. err. = 0.1420 px. **Top Right:** Horn&Schunck, av. err. = 0.0821 px. **Bottom Left:** 2nd Order Regularization, av. err. = 0.0525 px. **Bottom Right:** Optical Stokes Flow Computation, av. err. = 0.0480 px.

(ii) Navier-Stokes Flows

In the forthcoming examples, the assumption of a Stokes flow is definitely not valid, as these turbulent flows are mainly governed by the convection term of the Navier-Stokes equations. Therefore, we cannot expect the pressure or body-force to be physically accurate. However, our approach can also be used for these high Reynolds numbers. The Stokes equation then merely serves as a regularization term, and the body-force can be chosen so as to mimic the nonlinear effects of the convection term.

Cemagref Synthetic Highly Non-Rigid Image Pair This highly non-rigid synthetic PIV image pair was provided by [CH]. The underlying velocity field was computed by a so-called pseudo-spectral code that solves the vorticity transport equation in Fourier space, and evaluates a sub-grid model for simulating small-scale turbulent effects on the larger scales of the flow. The synthetic image intensity function was generated as in the preceding cases (sec. 5.3.3), its size is 256×256 pixels. The maximum displacement is approximately 3.5 pixels. We want to analyze this image pair using the following approaches:

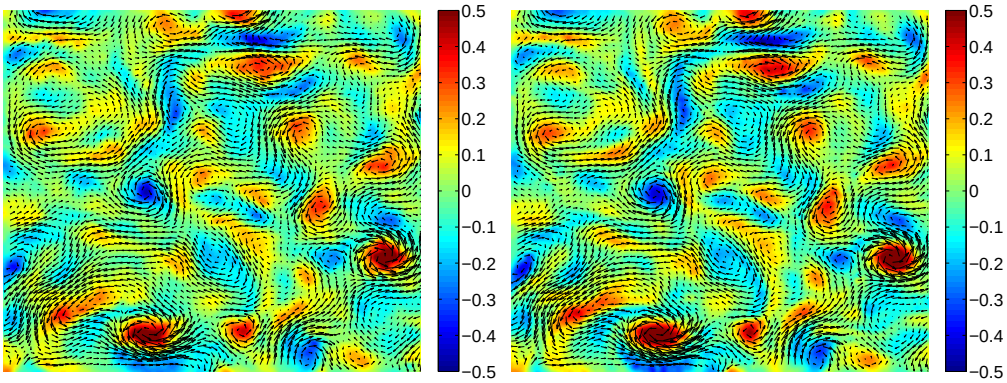

Figure 5.23.: Estimated velocity field and its curl. **Left:** Cross correlation. **Right:** Optical Stokes Flow. Note that cross-correlation slightly underestimates the vorticity.

- *Multi Pass Cross Correlation [LaV05]*: Advanced cross-correlation approach (LaVision Davis 7.1.1.34). Initial interrogation window size 32×32, final interrogation window size 8×8, and 50% overlap manually selected for best performance. In order to interpolate the velocity vectors to the fine grid (i.e. one vector per pixel), second-order spline interpolation is used.

- *Horn& Schunck [RKNS05]*: First-order regularization, no incompressibility constraint is imposed (cf. (5.12)). The smoothness parameter $\lambda = 0.005$ was manually selected for best performance.

- *2nd Order Regularization [YRMS05]*: The authors use higher-order regularization (cf. (5.13)) with an additional incompressibility constraint. Instead of mixed finite elements (that we use), the authors use the so-called mimetic finite differencing scheme. Temporal coherency is not exploited. Parameters: $\lambda_1 = 0.5$, $\lambda_2 = 0.05$, manually selected for best performance.

- *Optical Stokes Flow Computation (this paper)*: $\mu = 1$, $\alpha = 0.001$, $\gamma = 0.002$ (selected by hand).

Figure 5.22 shows the spatial error distributions for the different algorithms. Note that all variational approaches are able to outperform the cross-correlation method. Optical Stokes Flow computation yields the best results (average RMS error = 0.0484 pixels). Figure 5.23 compares the vorticity estimates for the cross-correlation approach and Optical Stokes Flow estimation.

Figure 5.24 shows how well the individual approaches are capable of recovering the different frequencies of the vorticity. While the spectrum of the Optical Stokes Flow estimate resembles the true spectrum very well, cross-correlation seems to underestimate higher frequencies. This had to be expected since cross-correlation relies on the assumption that the velocity gradient within an interrogation window is negligible. Even advanced window deformation techniques cannot fully resolve the high frequencies that are present in the velocity field.

It is interesting to note that Optical Stokes Flow gives extremely good results even though its prior knowledge is inadequate. In order to understand this fact, let us first summarize what type

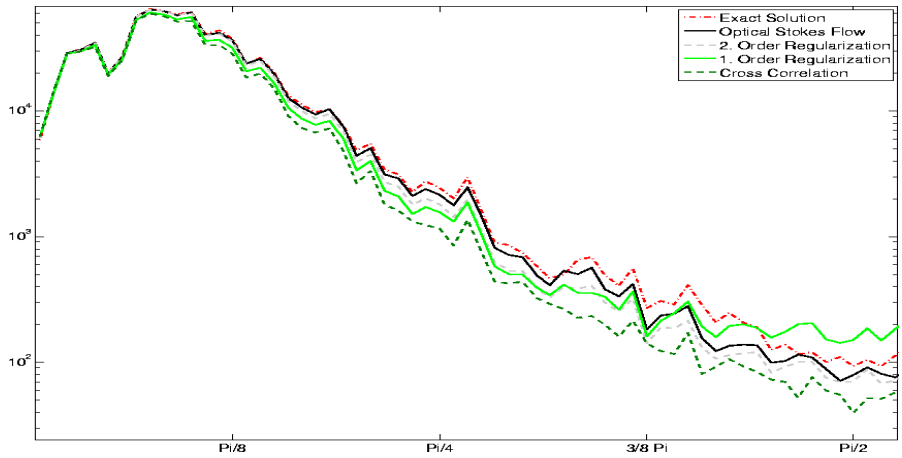

Figure 5.24.: Energy of the vorticity (for frequencies between 0 and $\Pi/2$) of the different algorithms (logarithmic plot). Energy spectrum of Optical Stokes Flow estimate is closest to the true solution, while cross-correlation underestimates higher frequencies.

of prior knowledge the competing approaches use:

- The cross-correlation approach assumes that the velocity field is piecewise constant. This assumption is weakened by advanced window-deformation techniques.
- The Horn&Schunck method assumes small velocity gradients.
- Higher-order div curl regularization assumes a smooth vorticity.

This compilation shows that the priors of all analyzed approaches are severely inadequate. It is clear that Optical Stokes Flow is only the starting point towards physically more and more plausible regularizers. In section 5.4, we will introduce a regularizer that uses the *full* Navier-Stokes equations as prior knowledge.

Highly Non-Rigid Real World Image Pair Figure 5.25 shows a sample image of the experimental evaluation of the spreading of a low diffusivity dye in a 2D turbulent flow, forced at a large scale. In contrast to the preceding examples, no tracer particles were brought into the fluid but a mixture of fluorescein and water. For more details about the experimental setup, we refer to [JCT01]. Cross-correlation approaches are not able to extract valid velocity fields for this type of input data (passive scalar images). Figure 5.25 shows, however, that our Optical Stokes Flow approach is capable of extracting a very reasonable velocity distribution. We chose the same parameters as in sec. 5.3.3.

Out-of-Plane Motion: Separation Bubble The synthetic image sequence that is shown in fig. 5.26 was generated by means of the software prescribed in [HK05b, HK05a]. Determination of the particle image displacements is based on the solution of a direct numerical simulation (DNS) of a laminar separation bubble.

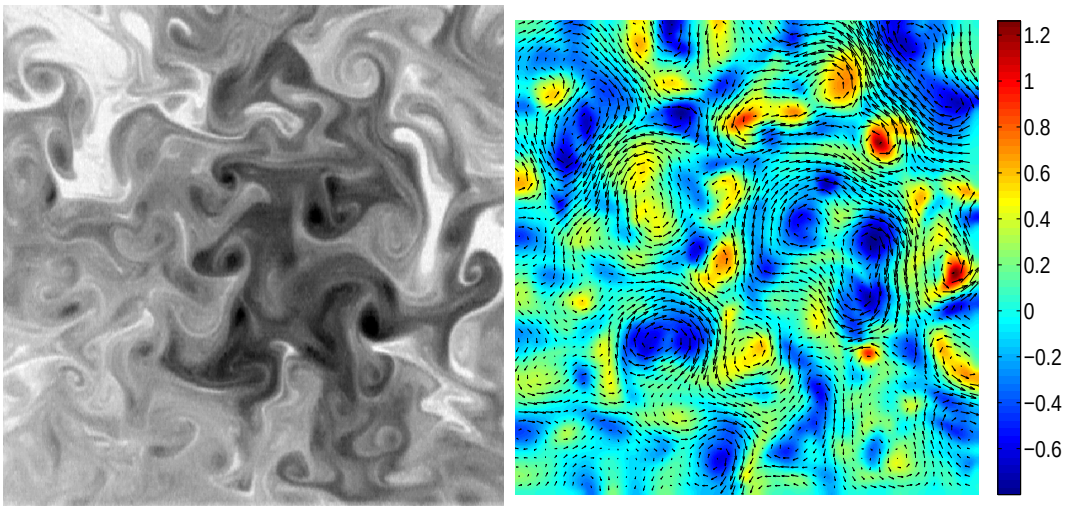

Figure 5.25.: **Left:** Passive Scalar Image (size: 512×512.) **Right:** Recovered Velocity Field using Optical Stokes Flow ($\mu = 1$, $\alpha = 0.001$, $\gamma = 0.002$). Note that cross-correlation approaches completely fail for this type of image data.

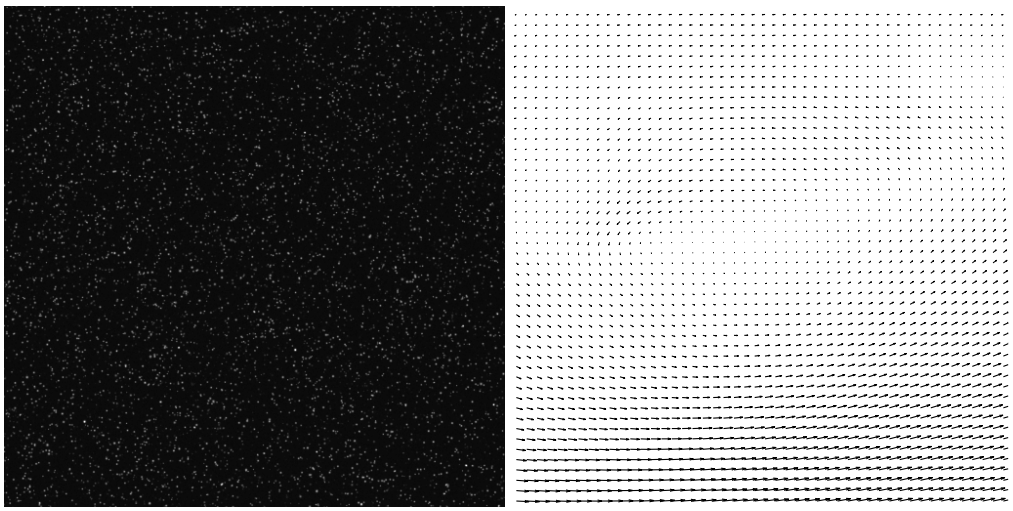

Figure 5.26.: **Left:** Synthetic image (Separation Bubble, size: 512×512.) **Right:** Synthetic velocity field. Note that the velocity field is three-dimensional. Therefore particles leave and enter the illuminated image plane. Furthermore, the 2D projection is no longer divergence-free.

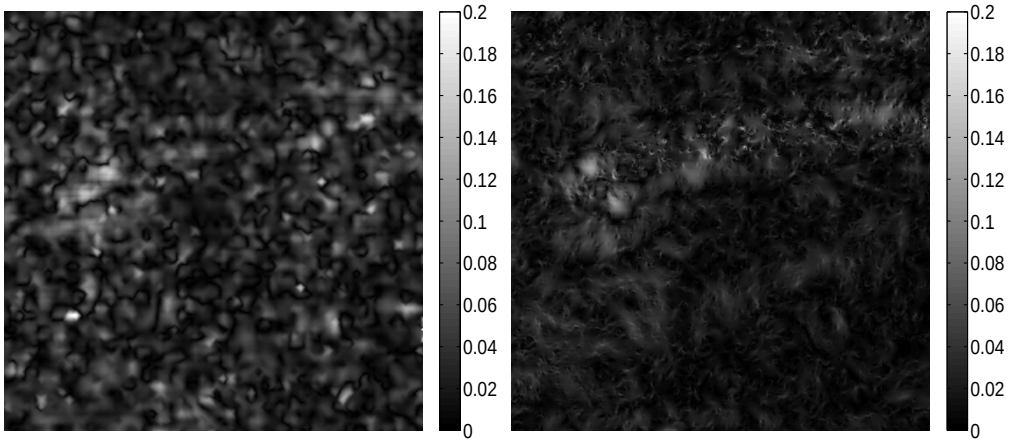

Figure 5.27.: **Left:** RMS velocity error of cross-correlation approach (smallest window size 16×16, 50% overlap, mean RMS = 0.0331 px.). **Right:** RMS velocity error of modified Optical Stokes Flow ($\mu = 1$, $\alpha = 0.001$, $\beta = 0.002$, mean RMS = 0.0240 px.).

The main problem in this example is the fact that the true velocity field of this sequence is three-dimensional (maximum out-of-plane velocity: 1 px.[8]). Because we analyze a 2D projection, the divergence-free constraint, that is strictly enforced by the Stokes equation, is not valid (cf. fig. 5.28). This is why we use the modified method of sec. 5.3.1 to analyze this image pair. Figure 5.27 compares the error of the cross-correlation method with our modified Stokes equation's velocity error. While cross-correlation yields an average error of 0.0331 pixel, the velocity field recovered by the modified Stokes approach is more reliable. Its average RMS error is 0.0240 pixels.

5.3.4. Conclusion

We presented a novel variational flow control approach for PIV that uses the Stokes equation as prior knowledge. Methods from flow control were used to solve the arising constrained optimization problem. The experimental evaluation showed that, as long as we confine ourselves to flows that are actually governed by the Stokes equation, the proposed algorithm is not only capable of reliably estimating the velocity fields between image pairs, but it can also extract the pressure distribution and forces acting on the fluid.

The experiments also showed that our approach is able to outperform other optical-flow-based methods as well as cross-correlation methods on highly non-rigid (Navier-Stokes) flows. The reason for this is the high resolution that can be achieved. We have proved that optical flow based approaches not only yield dense vector fields – with proper regularization (i.e. regularization that does not penalize velocity gradients) these dense vector fields excel the spatial resolution of cross-correlation approaches.

We have introduced a slight modification that enables the successful application of our approach also in (more realistic) scenarios, where the out-of-plane velocity is not negligible.

The most dramatic improvement that Optical Stokes Flow offers is when it comes to the

[8] We assume that the (imaginary) grid in out-of-plane direction has the same resolution as the in-plane grid.

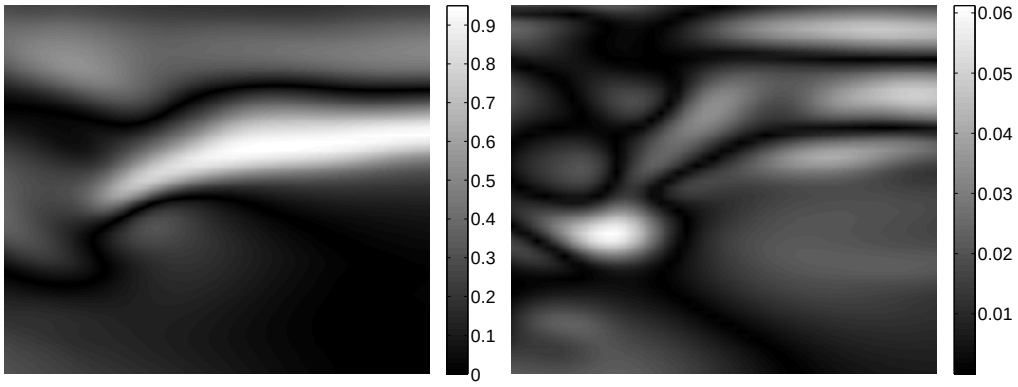

Figure 5.28.: Out-of-plane velocity (**Left**) leads to divergence of the 2D projection of the velocity field (**Right**). We have to use the modified Optical Stokes Flow approach to reliably reconstruct the target velocity field.

evaluation of scalar image pairs: Cross-correlation approaches completely fail on these image pairs, while optical flow- based methods can reliably extract velocity fields.

To summarize, the use of Optical Stokes Flow might be advantageous in the following three scenarios:

- Stokes flows: If the flow is actually governed by the Stokes equations, not only the velocity can be estimated but also pressure and forces that act on the fluid.
- Highly non-rigid flows: The increased spatial resolution that Optical Stokes Flow offers, permits the estimation of high-resolution velocity fields.
- Scalar Images: While cross-correlation approaches fail for this specific kind of image data, optical flow approaches give very reasonable velocity estimates.

5.4. Dynamic Motion Estimation with the Vorticity Transport Equation

Now that we have analyzed a regularizer that is based on the Stokes equation for creeping flows, we want to consider regularization terms that are better suited for high Reynolds numbers flows.

We present a framework for fluid motion estimation that uses as prior knowledge the fact that flows have to satisfy the incompressible vorticity transport equation. This equation relates to the full (incompressible) Navier-Stokes equations and is therefore also valid in *turbulent* scenarios. Furthermore, rather than considering image pairs, our estimation scheme takes into account the whole image sequence. As a result, it takes into account previous estimation results so as to enforce spatio-temporal coherency and regularization, *without*, however, penalizing flow structures that are characteristic for instationary turbulent flows. Finally, analogously to the corresponding concept from control theory, our overall algorithm works in a receding horizon manner, that is flow velocities can be computed as soon as their respective frames have been

recorded. In principle, this procedure sets the stage for the real-time extraction of instationary flow phenomena from particle image sequences.

In section 5.4.1, we first present the vorticity transport equation, which embodies the prior knowledge we use for flow estimation. Then, we motivate and describe our variational approach and details of the resulting constrained optimization problem. Corresponding numerical issues are dealt with in section 5.4.2. Numerical experiments for evaluating the approach are presented in section 5.4.3. We conclude in section 5.4.4.

5.4.1. Approach

The Vorticity Transport Equation

Let $u = (u_1, u_2)^\top$, $u = u(x,t)$, $x = \big(x_1(t), x_2(t)\big)^\top$, denote a two-dimensional velocity field. The incompressible vorticity transport equation is a specific form of the Navier-Stokes equation for homogeneous flow and can be expressed as follows:

$$\frac{D\omega}{Dt} = \frac{\partial}{\partial t}\omega + u \cdot \nabla\omega = \nu\Delta\omega \;, \quad \omega(x,0) = \omega_0 \;. \tag{5.33}$$

This equation is known as the *vorticity transport equation*. It describes the evolution of the fluid's vorticity over time. Note that in the absence of external forces acting on the fluid, this equation describes the flow completely. For a more detailed derivation of the vorticity transport equation, we refer to appendix C.3.

Variational Model

Let $I(x_1, x_2, t)$ denote the gray value of an image sequence recorded at location $x = (x_1, x_2)^\top$ within some rectangular image domain Ω and time $t \in [0, T]$. We adopt the basic assumption underlying most approaches to motion estimation, that I is conserved. Thus, the total (material) derivative of I vanishes:

$$\frac{DI}{Dt} = u \cdot \nabla I + I_t = 0 \;. \tag{5.34}$$

The spatial and temporal derivatives of I of the optical flow constraint (5.34) are estimated locally by using FIR filters (cf. sec. 4.1).

We have already seen (cf. sec. 3.4) that eqn. (5.34) alone cannot be used to reconstruct the velocity field u, because *any* vector field with components $u \cdot \nabla I = -I_t$ at each location x satisfies (5.34).

The standard approach is to minimize the squared residual of (5.34) over the entire image domain Ω, and to add a variational term that either enforces smoothness of the flow (*first-order* regularization) or smoothness of the divergence and vorticity (*second-order* regularization). In sec. 5.3, we have made a first attempt at physically plausible vector-field reconstruction by using the Stokes equation as a regularizer.

We emphasize that all these approaches take only into account *spatial* context and determine a vector field for a *fixed* point in time $t \in [0, T]$.

Therefore, following the ideas of [SRS06], we attempt to elaborate a *dynamic* representation of fluid flow. To this end, we solve eqn. (5.33) for the time interval $[0, T]$ between a subsequent pair of image frames, where ω_0 denotes our current vorticity estimate. As a result, we obtain a *transported* vorticity field $\omega_T := \omega(x, T)$, which can be regarded as a *predicted* vorticity, based

on the assumption that our fluid is governed by the Navier-Stokes equation. The regularization term that we employ penalizes derivations from the predicted vorticity values, and forces incompressibility:

$$\frac{1}{2}\int_\Omega \Big\{ \big(u \cdot \nabla I + I_t\big)^2 + \lambda \big(\nabla \times u - \omega_T\big)^2 \Big\} dx \;, \\ \text{s.t.} \quad \nabla \cdot u = 0 \;.$$

We apply Neumann boundary conditions (i.e. $\partial u / \partial n = 0$ on $\partial\Omega$). Note that, while the regularization term of (5.35) penalizes deviations between the current vorticity estimate ω and the propagated vorticity estimate of the preceding frame ω_T, it does *not* enforce smoothness of the current vorticity. In practice, an implementation of (5.35) therefore leads to increasingly noisy vorticity estimates. Increasing the parameter ν reduces the problem only slightly: ω_T becomes smoother, but smoothness of ω is still not enforced directly.
To overcome this problem, we add a term that mimics the small viscous term (Laplacian) on the right-hand side of eqn. (5.33). Expressing the new second-order regularization term equivalently through a first-order regularizer and an additional linear constraint, we finally obtain:

$$E = \frac{1}{2}\int_\Omega \Big\{ \big(u \cdot \nabla I + I_t\big)^2 + \lambda (\omega - \omega_T)^2 + \kappa |\nabla \omega|^2 \Big\} dx \;, \\ s.t. \;\; \nabla \cdot u = 0 \;, \\ \nabla \times u = \omega \;. \tag{5.35}$$

As we usually do not have a vorticity estimate at the very first frame of an image sequence, the overall estimation process is initialized with a vorticity estimate $\omega_0 = 0$.

The novel vorticity transport regularizer in (5.35), in connection with (5.33), can be perceived as a *special second-order div-curl regularizer*: Estimated flows from a given image sequence have vanishing divergence and a curl field (vorticity) that should be smooth and as close as possible to the transported vorticity.

5.4.2. Discretization and Optimization

Discretization of the Vorticity Transport Equation

We solve the time-dependent vorticity transport equation (5.33) with a second-order conservative finite difference algorithm. The method is upwind and two-dimensional, in that the numerical fluxes are obtained by solving the characteristic forms at cell edges (i.e. edges between adjacent pixels), and all fluxes are evaluated and differenced at the same time. The finite difference method that we employ is the Fromm-Van-Leer scheme [PC05].

The basic idea is to satisfy Godunov's theorem in a "natural" way. Roughly speaking, Godunov's theorem says that all methods of accuracy greater than order one will produce spurious oscillations in the vicinity of large gradients, while being second-order accurate in regions where the solution is smooth. Accordingly, Fromm-Van-Leer's scheme detects discontinuities and adapts its behavior, so that the high-order accuracy of Fromm's scheme is preserved for smooth parts of the solution, while spurious oscillations are avoided through first-order accuracy at detected discontinuities. For further details, we refer to appendix C and [PC05].

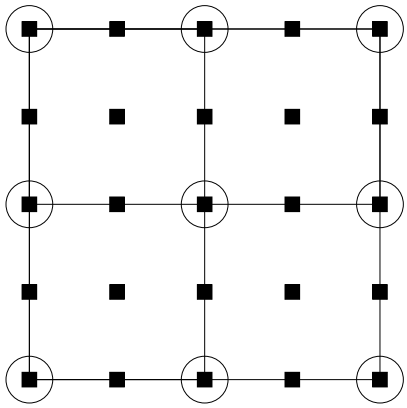

Figure 5.29.: Sketch of 2D Taylor-Hood elements: biquadratic velocity elements (squares) and bilinear pressure elements (circles).

Variational Approach

For every image pair (two consecutive frames of the image sequence), we have to solve optimization problem (5.35) which comprises a convex functional and two linear constraint equations. We transform this constrained optimization problem into a saddle-point problem. Accordingly, the unique vector field $u(x)$ minimizing (5.35), along with the vorticity ω and multipliers p, q, are determined by the variational system

$$\begin{aligned} a\big((u,\omega)^\top,(\tilde{u},\tilde{\omega})^\top\big) + b\big((p,q)^\top,(\tilde{u},\tilde{\omega})^\top\big) &= \big((f,g)^\top,(\tilde{u},\tilde{\omega})^\top\big)\ , \quad \forall \tilde{u},\tilde{\omega} \\ b\big((\tilde{p},\tilde{q})^\top,(u,\omega)^\top\big) &= 0\ , \quad \forall \tilde{p},\tilde{q}\ . \end{aligned} \tag{5.36}$$

The bilinear and linear forms read:

$$\begin{aligned} a\big((u,\omega)^\top,(\tilde{u},\tilde{\omega})^\top\big) &:= \int_\Omega \Big\{ u\cdot\nabla I\nabla I\cdot\tilde{u} + \lambda\omega\tilde{\omega} + \kappa\nabla\omega\cdot\nabla\tilde{\omega}\Big\}dx\ , \\ b\big((p,q)^\top,(\tilde{u},\tilde{\omega})^\top\big) &:= -\int_\Omega \Big\{ p\nabla\cdot\tilde{u} + q(\nabla\times\tilde{u}-\tilde{\omega})\Big\}dx\ . \end{aligned} \tag{5.37}$$

The right-hand side reads:

$$\big((f,g)^\top,(\tilde{u},\tilde{\omega})^\top\big) := \int_\Omega \Big\{ -I_t\nabla I\cdot\tilde{u} + \lambda\,\omega_T\tilde{\omega}\Big\}dx\ .$$

We choose a regular tessellation of the image domain Ω and discretize (5.36) using finite elements. It is well-known from computational fluid dynamics (cf. Stokes equation) that standard first-order finite element discretizations of saddle-point problems may result in instabilities or even in so-called locking effects, where the zero velocity field is the only one satisfying the incompressibility condition.

Therefore, when solving saddle-point problems, mixed finite elements are traditionally used [BF91]. An admissible choice is the so-called Taylor-Hood element based on a square reference element with nine nodes (fig. 5.29). Each component of the velocity field is defined in terms of piecewise quadratic basis functions ψ_i located at each node (the solid squares in fig. 5.29),

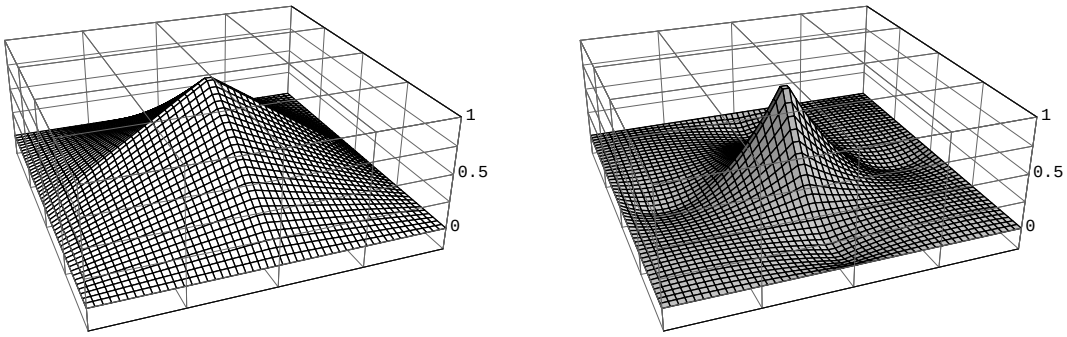

Figure 5.30.: **Left:** Basis function ϕ of a bilinear finite element. **Right:** Basis function ψ of a biquadratic finite element.

whereas the Lagrange multipliers p and q and the vorticity ω are represented by linear basis functions ϕ_i attached to each corner node (indicated by circles in fig. 5.29). It can be shown that Taylor-Hood elements fulfill the so-called Babuska-Brezzi condition [BF91], making the discretized problem well-posed. Appendix B goes further into saddle-point problems and their discretization with mixed finite elements.

Indexing the velocity nodes (squares in fig. 5.29) by $1, 2, ..., N$, we obtain

$$u_1(x) = \sum_{i=1}^{N} u_1^i \psi^i(x)$$

and similarly for $u_2(x)$ and the components of $\tilde{u}$. By analogy, we obtain for the M Lagrange multiplier nodes (circles in fig. 5.29)

$$p(x) = \sum_{i=1}^{M} p^i \phi^i(x)$$

and similarly expressions for $q, \omega, \tilde{p}, \tilde{q}, \tilde{\omega}$. Hence, each function $u, \tilde{u}$ is represented by $2N$ real variables, and each function $p, q, \omega, \tilde{p}, \tilde{q}, \tilde{\omega}$ is represented by M real variables. For the sake of simplicity, we will use the same symbols to denote the corresponding vectors. The discretized system (5.36) then reads

$$\begin{aligned} A(u,\omega)^\top \cdot (\tilde{u},\tilde{\omega})^\top + B^\top (p,q)^\top \cdot (\tilde{u},\tilde{\omega})^\top &= (f,g)^\top \cdot (\tilde{u},\tilde{\omega})^\top , \quad \forall \tilde{u},\tilde{\omega} \\ B(u,\omega)^\top \cdot (\tilde{p},\tilde{q})^\top &= 0 , \quad \forall \tilde{p},\tilde{q} . \end{aligned}$$

Because these equations have to be satisfied for *arbitrary* $\tilde{u}, \tilde{p}, \tilde{q}, \tilde{\omega}$, we finally obtain:

$$\begin{aligned} A\begin{pmatrix} u \\ \omega \end{pmatrix} + B^\top \begin{pmatrix} p \\ q \end{pmatrix} &= \begin{pmatrix} f \\ g \end{pmatrix} \\ B\begin{pmatrix} u \\ \omega \end{pmatrix} &= 0 \end{aligned} \tag{5.38}$$

In order to numerically solve the saddle-point problem (5.38), we want to employ the Uzawa algorithm (cf., e.g. [Bra97]). However, this requires A to be positive definite which is not the

case here, because the relations u and ω defining A in (5.37) are mutually independent, and u is only involved through a degenerate quadratic form. This problem can be removed by

- including a penalty term related to the divergence constraint into our Lagrange multiplier formulation to obtain an Augmented Lagrangian formulation [FG83], and by
- splitting the vorticity matching term into two equivalent terms, one containing $\nabla \times u$, and the other one containing ω.

This yields the following modification of the bilinear form (5.37):

$$\begin{aligned} a_p\big((u,\omega)^\top,(\tilde{u},\tilde{\omega})^\top\big) := \int_\Omega \Big\{ & u\cdot\nabla I\nabla I\cdot\tilde{u} + \frac{\lambda}{2}\big(\omega\tilde{\omega} + (\nabla\times u)(\nabla\times\tilde{u})\big) \\ & + \mu(\nabla\cdot u)(\nabla\cdot\tilde{u}) + \kappa\nabla\omega\cdot\nabla\tilde{\omega}\Big\} dx \ . \end{aligned} \tag{5.39}$$

We point out that this modification is done for numerical reasons only. It does not change the optimization problem (5.35). Matrix A_p resulting from the discretization of (5.39) is positive definite and, because u and ω do *not explicitly* depend on each other, can be split into two systems:

- The system containing u is the linear system with a simple first-order div-curl regularization (cf., e.g. [Sut93], and (5.4)).
- The system containing ω corresponds to a simple first-order quadratic functional.

Because A_p is invertible and well-conditioned, we solve the first equation of the system (5.38), with A replaced by A_p, for the unknown $(u,\omega)^\top$

$$\begin{pmatrix} u \\ \omega \end{pmatrix} = A_p^{-1}\left[\begin{pmatrix} f \\ g \end{pmatrix} - B^\top \begin{pmatrix} p \\ q \end{pmatrix}\right] \ ,$$

and insert the result into the second equation:

$$BA_p^{-1}\left[\begin{pmatrix} f \\ g \end{pmatrix} - B^\top \begin{pmatrix} p \\ q \end{pmatrix}\right] = 0 \ .$$

This problem only involves the adjoint variables p, q:

$$(BA_p^{-1}B^\top)\begin{pmatrix} p \\ q \end{pmatrix} = BA_p^{-1}\begin{pmatrix} f \\ g \end{pmatrix} \ . \tag{5.40}$$

The matrix $(BA_p^{-1}B^\top)$ is symmetric and positive definite. Therefore, we apply the conjugate gradient iteration to (5.40). This requires a single matrix inversion in every iteration step. For computational efficiency, this is accomplished using multi-grid iteration (cf. [Hac93]).

Weakening the Assumption of a Vanishing Divergence

Due to out-of-plane motion (that can hardly be totally avoided), the assumption of a vanishing divergence will usually not hold in practice.

Let us therefore weaken the assumption and minimize

$$E = \frac{1}{2}\int_\Omega \Big\{(u\cdot\nabla I + I_t)^2 + \lambda(\omega-\omega_T)^2 + \kappa|\nabla\omega|^2 + \mu|d|^2\Big\}dx\ ,$$
$$s.t. \qquad \nabla\cdot u = d\ ,$$
$$\nabla\times u = \omega\ ,$$

where the 2D divergence d (which is assumed to be small) does actually correspond to the derivative of the out-of-plane component of u in out-of- plane direction (i.e. $d = \partial u_3/\partial z$). Note that we do not change the vorticity tranport equation itself – we still assume that the 2D vorticity transport equation is able to give a good approximation for the transport process. Therefore, (5.41) should only be used to analyze 2D projections of incompressible fluids.

5.4.3. Experimental Evaluation

This experimental section is divided into three parts:

- Firstly, we present numerical results on ground-truth fluid image sequences (2D flows) obtained with our approach, in comparison with cross-correlation and optical flow with first-order and with higher-order regularization.
- Secondly, we show numerical results for a synthetic flow where the out-of-plane component is not negligible (3D flow). We perform the analysis with the method introduced in sec. 5.4.2.
- Thirdly, we show results for a real-world 2D image sequence.

Synthetic 2D Flows

This section shows numerical results on ground-truth fluid-image sequences obtained with our approach, in comparison with cross-correlation and optical flow with first-order regularization and higher-order regularization.

The synthetic PIV image sequence that we used for testing was provided by [CH]. The underlying velocity field was computed by a so-called pseudo-spectral code that solves the vorticity transport equation in Fourier space, and evaluates a sub-grid model for simulating small-scale turbulent effects on the larger scales of the flow. These latter effects, of course, are *not* known in practice, nor was anything related to these effects used while evaluating our approach.

In order to simulate the intensity function of real PIV images, the computed velocity fields are used to transport collections of (images of) particles that are typically used for the seeding of flows, so as to make them visible. The scheme resembles the one described in [ONK00a]. We used the first 100 frames of the synthesized PIV image sequence and compared the following three approaches:

- *Multi-Pass Cross-Correlation [LaV05]:* Advanced cross-correlation approach (LaVision Davis 7.1.1.34). Initial interrogation window size 32×32, final interrogation window size 8×8, and 50% overlap, manually selected for best performance. In order to interpolate the velocity vectors to the fine grid (i.e. one vector per pixel), second order spline interpolation is used.

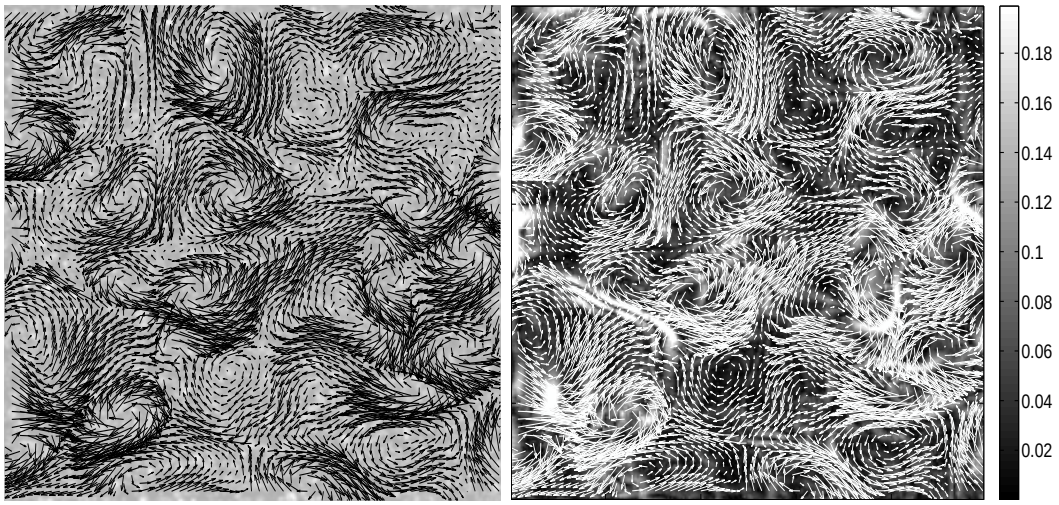

Figure 5.31.: **Left:** 100th frame of the synthetic image sequence with ground truth velocity field. **Right:** Estimated velocity field for the 100th frame. The background intensity shows the absolute RMS error (brighter = larger error), which is about 0.055 px. on average (cf. fig. 5.32).

- *Horn&Schunck (cf. sec. 3.4):* First-order regularization; temporal coherency is not exploited; no incompressibility constraint is imposed. The smoothness parameter $\lambda = 0.005$ was manually selected for best performance.
- *2nd Order Regularization (cf. sec. 5.1.2):* The authors used higher-order regularization with an additional incompressibility constraint. Instead of mixed finite elements (that we used), the authors used the so-called mimetic finite differencing scheme. Temporal coherency is not exploited. Parameters: $\lambda_1 = 0.5$, $\lambda_2 = 0.05$, manually selected for best performance.
- *Optical Stokes Flow (cf. sec. 5.3):* Optical flow approach that incorporates physical prior knowledge. Admissible flow fields are restricted to vector fields satisfying the Stokes equation. Parameters: $\mu = 1$, $\alpha = 0.001$, $\gamma = 0.002$, manually selected for best performance. Temporal coherency is not exploited.
- *Vorticity Transport Approach (this section):* As described above, higher-order regularization is used, the incompressibility constraint is imposed, and temporal coherency is exploited in an on-line manner. Parameters $\lambda = 0.005$, $\mu = 0.0025$, $\nu = 1$, $\kappa = 0.0005$. As for the other approaches, we selected the regularization parameters λ, μ, κ by hand. Note that the viscosity coefficient ν is not a free user parameter but characterizes the physical nature of the fluid flow.

Figure 5.32 compares the errors of all five approaches over time. The multi-pass cross-correlation approach's estimate has the highest RMS error. This is due to the very high velocity frequencies that are present in the image data and that cannot be recovered by correlation. First-order regularization yields a higher error than second-order regularization, which is much more accurate. The quality of the estimation can further be improved by applying Optical Stokes

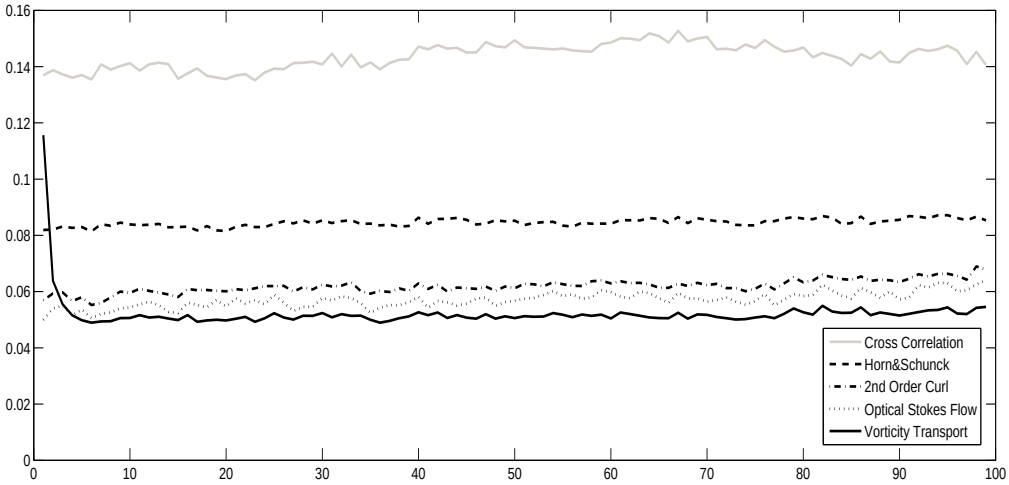

Figure 5.32.: Average absolute RMS error (in pixels) for frames 1-100, using five different methods. Cross-correlation gives the worst results for this highly non-rigid image pair. First-order regularization performs worse than second-order regularization, while Optical Stokes Flow is slightly better than second-order regularization. All these four error curves are constant because temporal coherency is not exploited. The approach based on vorticity transport starts with a rather low accuracy (assumption of $\omega = 0$, which is not valid) but then becomes significantly more accurate than the other techniques due to the physically consistent regularization over time. This novel spatio-temporal regularization is achieved with an on-line computational scheme and fixed storage requirements, irrespective of the length of the image sequence. The decay of the error curve within the first 10 frames clearly displays the usage of this implicitly encoded "memory".

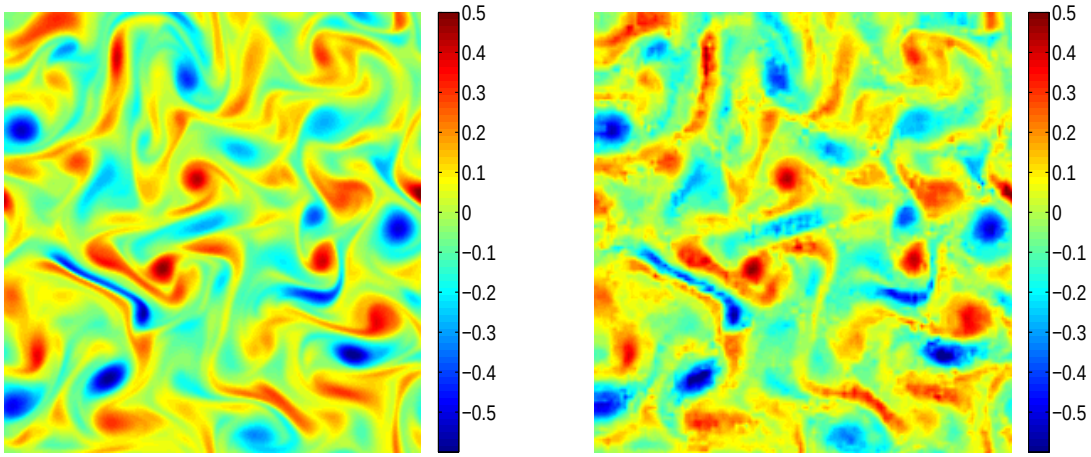

Figure 5.33.: **Left:** True vorticity of frame 100. **Right:** Estimated vorticity ω for frame 100. For the first frame, the estimation process was initialized with $\omega = 0$, corresponding to "nothing is known in advance". The result on the right shows that not only the vorticity transport equation has been successfully adapted to the observed image sequence, but that it improves the accuracy of flow estimation in terms of u, too (cf. fig. 5.32). As a consequence, flow *derivatives* can be estimated fairly accurate, as shown in the right panel. Such quantitative information is very important in connection with imaging-based experimental fluid mechanics.

Flow. The errors of all these four approaches stay constant over time because each subsequent image pair is independently evaluated, and temporal coherency is ignored.

For the first frame, the approach presented in this section, utilizing the vorticity transport equation, shows worse performance than the other optical-flow-based algorithms. During the subsequent period of time, however, the error of the vorticity transport approach decreases considerably, because not only higher-order regularization is used, but temporal coherency is successfully exploited as well.

We emphasize that temporal coherency does *not* mean smoothness. Rather, the flow exhibits high spatio-temporal gradients as turbulent fluids do. Temporal coherency relates to a physically consistent transport mechanism interacting with flow estimation from an image sequence. Due to the on-line computational scheme, fixed computational resources are needed no matter how long the image sequence is. The decay of the error curve over several frames in figure 5.32 shows, however, that the approach is able to memorize the history longer than just the previous frame.

Figure 5.31 displays the estimated velocity for the 100th frame, along with the respective RMS errors. The reconstructed velocity field is surprisingly exact, in view of the highly non-rigid motion we are dealing with. Figure 5.33 shows that even the vorticity related to flow *derivatives* is reconstructed quite well under these difficult conditions. We expect such quantitative data to be valuable information in connection with imaging-based fluid mechanics.

Flows with Out-of-plane Velocity Component

In order to assess our approach's performance when it comes to tackling image sequences with a high out-of plane component, we analyzed the VSJ image sequence 301 [ONK+00b]: It shows a 3D jet shear flow with an out-of-plane component of up to 4 pixels. [9] Due to the large out-of-plane velocity component, the assumption of a vanishing divergence does not hold in this example. This is why we weaken the assumption of a vanishing divergence, as shown in section 5.4.2.

Figure 5.34 compares the results of our vorticity transport approach with those of an advanced cross-correlation approach (DaVis). For the evaluation, we chose the following parameters:

- *Multi-Pass Cross-Correlation:* Initial interrogation window size 32×32, final interrogation window size 16×16, and 50% overlap, manually selected for best performance.
- *Vorticity Transport Approach:* $\lambda = 0.01$, $\mu = 0.005$, $\nu = 0.1$, $\kappa = 0.005$, manually selected for best performance.

Figure 5.34 shows the absolute RMS error of both approaches along with the average absolute out-of-plane motion over time. While both error curves are quite similar, the cross-correlation approach tends to give better results at time instances when the out-of-plane velocity is rather large (i.e. $t \approx 40$, and $t \approx 125$), whereas the optical flow results are better when the out-of-plane component is rather small (i.e. $t \approx 1$, and $t \approx 70$).

The fact that the brightness of particles that travel out of the illuminated plane will fade, while particles gain brightness if they travel towards the illuminated plane, is in contradiction with the optical flow constraint that we use. This problem introduces errors in scenarios where high out-of-plane velocities are present. We'd like to stress, however, that cross-correlation approaches have the same problem (as they also assume brightness conservation), it just seems to be slightly less pronounced.

Real-World 2D Flows

Figure 5.35 shows a sample image of the experimental evaluation of the spreading of a low diffusivity dye in a 2D turbulent flow, forced at a large scale. The passive scalar is a mixture of fluorescein and water. For more details about the experimental setup, we refer to [JCT01]. Cross-correlation approaches are not able to extract valid velocity fields for this type of input data (passive scalar images). Figure 5.35 shows, however, that our approach that uses the vorticity transport equation, is capable of extracting a very reasonable velocity distribution. Figure 5.36 shows the temporal evolution of individual vortices.

5.4.4. Conclusions

We presented an approach to fluid motion estimation that uses the vorticity transport equation for physically consistent spatio-temporal regularization. The approach combines variational motion estimation with higher-order regularization and motion prediction through a transport process. For motions that conform to our assumption (i.e. fluids that are governed by the incompressible 2D Navier-Stokes equation), a temporal regularization effect, computed in a recursive manner,

[9] Note that we assume that the imaginary grid in out-of-plane direction has the same resolution as the in-plane grid.

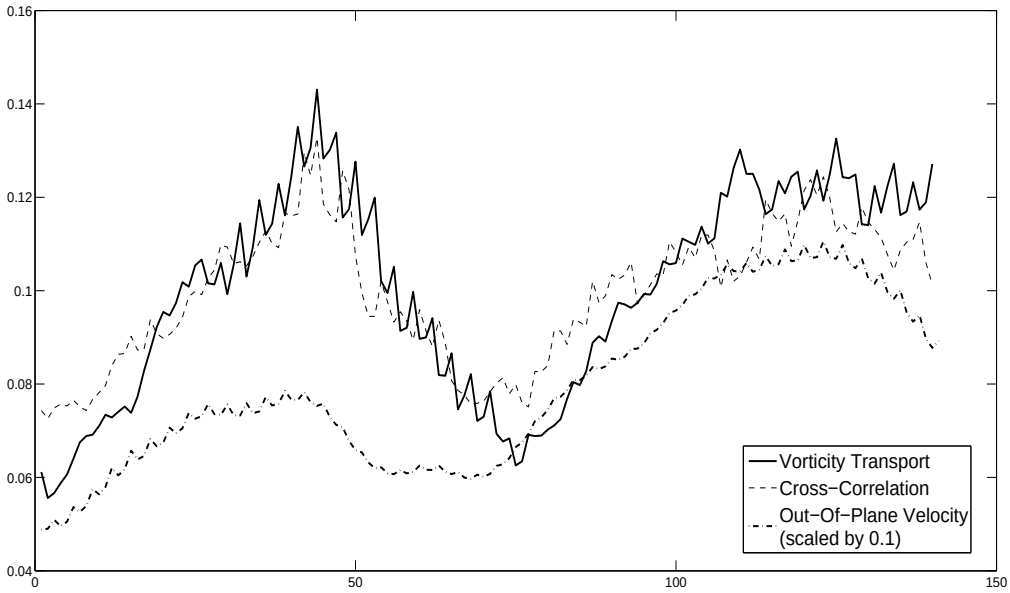

Figure 5.34.: Average absolute RMS error (in pixels) for frames 1-140 of the VSJ 301 image sequence, using cross-correlation and novel optical flow technique with spatio-temporal regularization (with modification of sec. 5.4.2). Both approaches have similar accuracy.

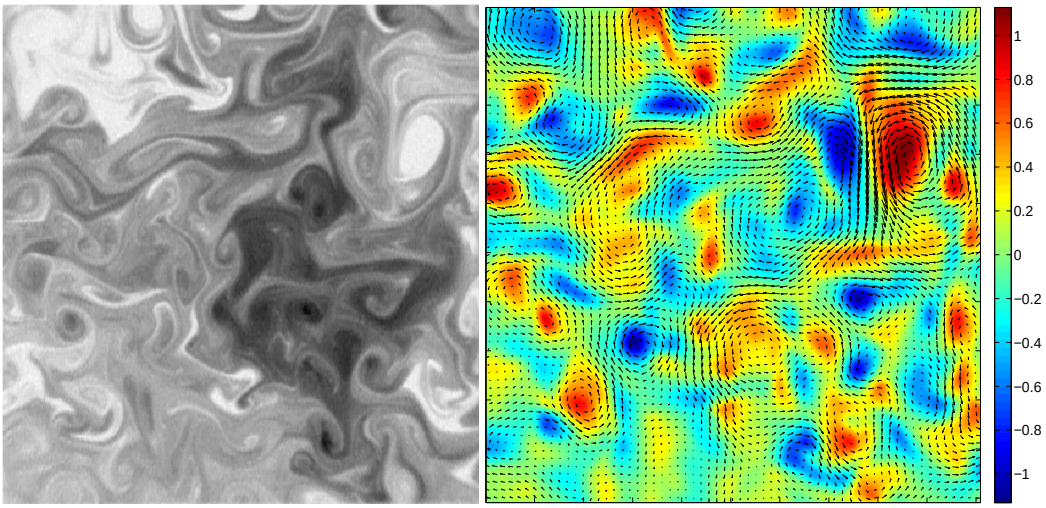

Figure 5.35.: **Left:** Sample Real-World Passive Scalar Image (frame 80, size: 512 $\times$ 512 px.). **Right:** Recovered Velocity Field (with color-coded vorticity) with Vorticity Transport Approach.

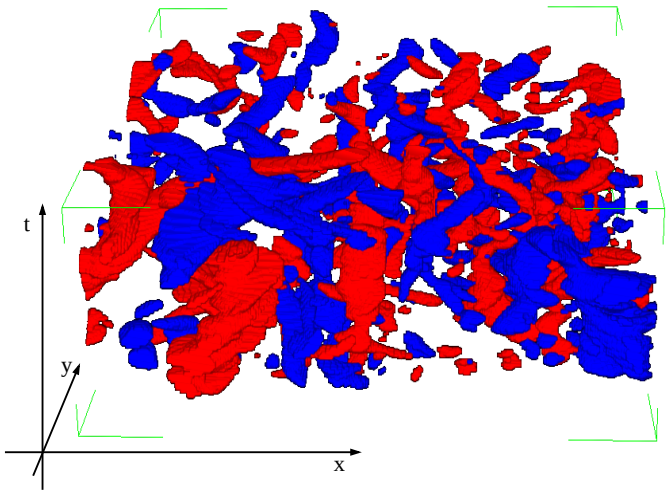

Figure 5.36.: Iso-surface plot of the vorticity distribution over time. Blue denotes positive vorticity ($\omega > 1.5$) and red denotes negative vorticity ($\omega < -1.5$).

was demonstrated. In these scenarios, our approach outperforms cross-correlation approaches as well as advanced variational approaches for optical flow estimation.

6. Conclusion

6.1. Summary

Let us shortly summarize the important points of this thesis. The main issue of this manuscript was to introduce a variational framework for the analysis of PIV images and sequences. In chapter 2, we have investigated standard cross-correlation and tracking algorithms. We have pointed out some limitations of these methods – the most important limitations are:

- The highest reachable resolution of cross-correlation PIV is limited. Due to the statistic nature of cross-correlation PIV, there is a tradeoff between interrogation window size and resolution of the recovered velocity estimates. While large windows lead to robust but coarse estimates, small windows are able to capture higher frequencies of the velocity fields at the cost of reduced robustness.
- Motion estimation if carried out regardless of spatial context. Prior knowledge about spatial or temporal coherency cannot be exploited directly, but has to be inferred by (heuristic) post-processing steps.
- Traditional PIV and PTV methods can only be applied to typical PIV imagery. If, e.g., a passive scalar field is to be analyzed, traditional cross-correlation or tracking methods will fail.

These were the main reasons that motivated the use of variational optical flow methods.

Variational Particle Image Velocimetry The mathematical basis of the methods presented in chapter 3 is a continuous variational formulation for globally estimating the optical flow vector fields over the whole image. Minimizing the respective variational functionals yields *dense* (i.e. one vector per pixel) velocity fields – there are no interrogation areas whose size might limit the resolution. The class of approaches had been known in the field of image processing and computer vision for more than two decades but apparently had not been applied to PIV image pairs so far. In sec. 4.1, we described the first steps of adapting the data term to the quite specific signal structure of particle image pairs:

- Due to changes in the illumination of PIV setups (that are often unavoidable), the assumption of gray value conservation (the traditional assumption behind optical flow estimation) is not valid. We adapted the prototypical optical flow constraint so that it tolerates these illumination changes.
- PIV imagery often consists of only image pairs, while standard variational approaches for motion estimation operate on whole image sequences. We reformulated the optical flow constraint and made it symmetric, and thus improved accuracy of optical flow estimation between image pairs.

- The typical motion range in PIV imagery is very high. It is crucial for PIV evaluation algorithms to be able to resolve the full range of motion. To handle large displacements, we presented a coarse-to-fine scheme.

The experimental evaluation showed that a prototypical variational approach competes in noisy real-world scenarios with four alternative approaches especially designed for PIV-sequence evaluation.

Variational Particle Tracking Velocimetry In sec. 4.2, we have expanded the class of variational methods to the field of particle tracking velocimetry: We introduced a novel variational approach for evaluating PTV image pairs and sequences in two and three dimensions. We combined a discrete non-differentiable particle matching term with a continuous regularization term. An advanced mathematical method guaranteed convergence to a local minimum. Our method has the following features:

- Like most competing approaches, we combine the two assumptions *small displacements* (from one frame to the next) and *smoothness of motion* (i.e. two particles that are neighbors in the first frame are likely to be neighbors in the second frame). In contrast to other approaches to PTV, we combine the two assumptions in a mathematically sound way by minimizing a variational functional.

- We showed that it posed no problem to expand our method to the analysis of even 3D image sequences (where the assumption of temporal coherency is also included). This expansion follows directly from expanding the involved functions from 2D to 3D (or 3D+time). The variational method and the corresponding FEM discretization directly appoint the equations that have to be solved.

- It is straightforward to replace the simple smoothness term that we used by physically motivated priors.

The experimental evaluation showed that our variational method competes with three alternative approaches.

Variational Motion Estimation with Physics-based Priors In chapter 5, we addressed the question of how to adapt the prototypical regularization term from chapter 3 to the quite specific demands of PIV velocity fields. We collected various possibilities of minimizing variational motion estimation functionals that incorporate higher order regularization terms to estimate more accurate velocity fields.

We argued that the physical plausibility of standard higher-order regularization terms is only very limited. In sec. 5.3, we therefore presented an approach to particle image velocimetry based on optical flow estimation subject to valid physical constraints. Admissible flow fields are restricted to vector fields satisfying the Stokes equation. The latter equation includes control variables that allow to control the optical flow, so as to fit to the apparent velocities of particles in a given image pair. We showed the following features of this so-called *Optical Stokes Flow* technique:

- When the real unknown flow observed through image measurements conforms to the physical assumption underlying the Stokes equation, the control variables allow for a physical interpretation in terms of pressure distribution and forces acting on the fluid.

- Although this physical interpretation is lost if the assumptions do not hold, our approach still allows for reliably estimating more general and highly non-rigid flows from image pairs.
- We made the important observation that optical flow estimates (together with a reasonable regularizer like the Optical Stokes Flow regularizer) are able to outperform cross-correlation methods in terms of achievable resolution and quality of the extracted velocity field.

We argued in sec. 5.4 that Optical Stokes Flow has two main drawbacks:

- The physical plausibility of Optical Stokes Flow is limited to cases of very low Reynolds numbers (as the approach neglects the convective terms of the Navier-Stokes equations).
- As Optical Stokes Flow uses the time-independent Stokes equation as prior knowledge, it cannot exploit temporal coherency.

This is why we extended the approach in sec. 5.4 along two directions:

- The full incompressible Navier-Stokes equation was employed in order to obtain a physically consistent regularization which does not suppress turbulent flow variations.
- Regularization along the time-axis was employed as well, but formulated in a receding horizon manner contrary to previous approaches to spatio-temporal regularization. This allowed for a recursive on-line (non-batch) implementation of our estimation framework.

Ground-truth evaluations for simulated turbulent flows demonstrated that due to imposing both physical consistency and temporal coherency, the accuracy of flow estimation compares favorably even with optical flow approaches based on higher-order div-curl regularization and advanced cross-correlation approaches.

6.2. Open Problems and Further Work

There are a several open problems that we will have to tackle in the future. In this section we shortly outline these problems and present some ideas about how to solve them.

Automatic Parameter Selection In this manuscript, we tried to introduce as few parameters as possible. The use of regularization parameters for the type of variational approaches that we use, is, however, nearly without alternative. In principle, the variational approaches can be rewritten as maximizing an a posteriori probability (with the use of the Bayes formula). However, the parameters of the individual probability distributions are generally not determinable – so *again*, they would have to be considered as user parameters. Though we have selected the best parameter settings by hand in most experiments, we have also accomplished a series of experiments (e.g. sec. 4.1.4, 4.2.3) that shows that a non-optimal parameter choice still yields good results. We agree, however, that some kind of automatic parameter selection process would be desirable. In [HNC+06], the authors go a first step on this way by introducing a non-dimensional regularization parameter (through dimensional analysis of the optical flow constraint) and by discussing the influence of the individual scales *luminance*, *space*, and *time*.

Individual Particle Detection for PTV We have seen in sec. 2.2.2, that PTV methods actually consist of (at least) two steps that are usually conducted in the following order:

- individual particle detection
- temporal matching of the individual particles

In the case of 3D PTV, the following step is introduced between the two steps above:

- 3D reconstruction trough triangulation

In section 4.2 we have only dealt with the temporal matching of individual particles. We have seen in section 4.3, however, that, in order to compare PIV and PTV approaches, one has to follow all the steps in order to be able to give typical error measures.

A further interesting point is a possible combination of the above mentioned points in one single optimization approach. Combining spatial matching (3D reconstruction) and temporal matching might improve the overall performance of the algorithm: The fact that temporal matching fails is a hint on an erroneous spatial matching. A functional that combines both steps would, however, be highly non-convex, and it would be hard to find a significant minimum.

Physically Motivated Priors for PTV In section 4.2, we have regularized our matching functional with the simple assumption of a smooth target velocity field. In chapter 5 we have presented regularizers that are much more adequate for fluid scenarios. These regularizers could easily be adapted for PTV, too. We believe that an adaptation of the spatio-temporal regularization introduced in section 5.4 for the purpose of PTV is especially promising: While the spatio-temporal regularization introduced in sec. 4.2 is an *Eulerian* approach (i.e. one assumes that the velocity at a fixed point should only vary smoothly over time), the regularizer of section 5.4 is of *Lagrangian* nature: One regularizes over a moving particle (i.e. with a moving coordinate system). It is clear that a "Lagrangian" regularization has the potential of yielding better results in turbulent scenarios.

Displacement vs. Velocity In the approaches presented in chapter 5, we assumed that velocity and displacement are identical: Note that the optical flow data term gives an estimate of the *velocity*. However, if we consider the same data term in a coarse-to-fine or iterated registration framework, the output of the iterative energy-minimization process is a *displacement*. In contrast, the regularizers that we have introduced in chapter 5 are valid for *velocities* only. We have bypassed addressing this problem, by only considering (in chapter 5) image pairs and sequences with small movements of the individual particles from one frame to the next. In these environments, one can use the terms displacement and velocity synonymously.

If one wants to apply physically plausible regularization in image pairs and sequences with large displacements, a good starting point might be the work of [GM98] and references therein. In [BMT05], the authors study a large deformation diffeomorphic metric mapping problem. They propose a variational formulation to estimate an optimal transformation between two images in the space of smooth *velocity* vector fields.

Optimal Spatio-Temporal Control We have seen that Optical Stokes Flow (cf. sec. 5.3) is an optimal spatial control approach. In section 5.4, we have expanded this approach to image sequence analysis and chose the vorticity transport equation as prior knowledge. Additionally, we demanded that on-line (non-batch) processing should be possible. At every time instance, our approach therefore chooses the most plausible velocity field (using the current image data and the resulting velocity field of the preceding frames).

As the approach uses (at every time instance) only image data from the "present" and the "past", this decision will usually not be "a posteriori" optimal: In scenarios in which the *whole* image sequence is *a priori* available, it makes sense to use this knowledge during optimization.

The optimal control approach introduced in sec. 5.3 could be expanded to image sequences, and an optimal solution could be computed in a forward-backward manner (cf. e.g. [BIK02, Gun02]).

Dangers of using (Physical) Priors In this thesis, we have used variational approaches for motion estimation because they allow to incorporate prior knowledge into the estimation of velocity vector fields. The most simple prior knowledge that we have used in chapter 4 was the assumption of a smooth target velocity field. In practical scenarios, problems might start at this point: Users of cross-correlation approaches can clearly detect from visual inspection (through the outlier percentage) whether a velocity field is plausible or not. If a cross-correlation approach fails (e.g. due to bad experimental conditions or a wrong parameter choice) one simply gets *noise* as the output velocity field. With variational approaches, results are somewhat different: If a variational approach is not able to reconstruct a meaningful velocity field, it will still reconstruct a vector field that (at least) minimizes the regularization term: Using smoothness as a prior, the reconstructed vector field *will be* smooth, and this fact might entrap the user to believe that the velocity field is reliable. We believe that analyzing the computed overall energy is a good starting point on the way to a reliable confidence measure.

Further problems occur through the use of physically based prior knowledge: Every piece of knowledge that is used, will bias the solution. If we, e.g., assume incompressibility of the fluid, then our algorithm will output a velocity field with zero divergence – independently from whether the assumption is true or not.

To summarize, one has to be extremely careful about what kind of prior knowledge one chooses. Before one introduces more and more advanced types of priors, one should discuss with fluid mechanics engineers and fluid experimentalists, whether these constraints are actually plausible.

Fluid Priors for Non-Fluid Scenarios An interesting open problem in computer vision is finding prior knowlege for motion estimation in *general* image sequences (e.g. human motion, traffic scenes, ...). In contrast to fluid imagery, there are no (rather) simple physically plausible priors available in these scenarios. Therefore, the problem is much more difficult. The use of fluid priors enables us to model simple physical facts (which are also true for general image sequences, e.g. inertia) with partial differential equations. A first step in this direction has been performed in [SRS06], where the authors model the scene as a fictive fluid that is governed by the Burgers equation (i.e. the acceleration of ficitive fluid particles is penalized).

From 2D to 3D Note that we have only expanded the PTV approaches to 3D, as yet. All gray-value-based approaches that we introduced in this thesis are restricted to 2D. Currently, there has been a lot of research published on experimental PIV methods, that yield volumetric gray value information. We believe that our variational approaches are especially suitable in these 3D scenarios:

- We have seen in section 3.2, that the optical flow constraint assumes gray value conservation. If particles move in or out of the illuminated plane, the optical flow constraint is violated and the error rates increase[1]. If we illuminate the whole volume, a 3D optical flow constraint is more likely to hold.
- Imposing physical constraints is much more straightforward in 3D than it is in 2D. Both in sec. 5.3.1 and 5.4.2, we had to handle out-of-plane motion separately – in 3D, this unattractive special treatment will become redundant.
- Due to the fact that in 3D experimental setups the resolution is often reduced [2], the possibility to include (physically motivated) prior knowledge is tempting.

[1] Note that this consideration is true for *any* kind of motion estimation algorithm (including cross-correlation based methods).

[2] Scanning PIV allows high resolution in two dimensions, but the sampling rate in the third dimension is usually much lower. The resolution of tomographic reconstruction algorithms is in turn limited by the fact that experimental conditions forbid the use of an arbitrary number of cameras.

A. Elliptic Systems and FEM

This section is to give a short overview on how to handle equations of elliptic type. We first give a very short introduction into elliptic theory and then we describe how to discretize these systems using finite elements. Finally we introduce the numerical methods that we use in this thesis to solve large systems of equations which emerge from finite element discretization. Note that our overview is rather algorithmical and far from being complete, for a sound mathematical introduction into elliptic systems and finite elements, we refer to [Cia78, Bra97].

A.1. Elliptic Theory

In this thesis, there are a number of problems (e.g. (3.12),(4.3),(4.12)), that demand minimizing some energy functional

$$\inf_{v \in H} J(v)$$

where H is a Hilbert space.
In the following, we assume that the functional $J(v)$ can be written as

$$J(v) = \frac{1}{2} a(v,v) - \langle f, v \rangle,$$

where $a : H \times H \to \mathbb{R}$ is a symmetric bilinear form and $f : H \to \mathbb{R}$ is a linear functional. In order to evaluate whether $J(v)$ has exactly one solution, we first have to introduce *continuity* and *H-ellipticity*.

Definition 1. *A bilinear form $a : H \times H \to \mathbb{R}$ is* **continuous** *if $\exists\, C < \infty$ with*

$$|a(u,v)| \leq C||u|| \cdot ||v||, \ \forall u, v \in H,$$

where H is a Hilbert space.

Definition 2. *A symmetric continuous bilinear form is* **H-elliptic***, if*

$$a(v,v) \geq \alpha ||v||^2, \ \forall v \in H,$$

with $\alpha > 0$.

Now we can present the Lax-Milgram theorem for convex sets:

Theorem 3. Lax-Milgram Theorem*: V is a closed convex set in a Hilbert space H and $a : H \times H \to \mathbb{R}$ is an H-elliptic bilinear form. For every $f \in H'$, the variational problem*

$$J(v) := \frac{1}{2} a(v,v) - \langle f, v \rangle \to \textit{min}$$

has exactly one solution.
For a proof of the Lax-Milgram theorem for convex sets, we refer to [Bra97].

We have seen that it is sufficient to show that a given symmetric bilinear form is continuous and H-elliptic. For the approach of Horn&Schunck (cf. (3.12)), this was performed in [Sch91].

A.2. Ritz-Galerkin Method

The most natural approach of solving elliptic problems lies in the use of finite elements. This goes back to [Rit08]. The minimum of a functional J of some variational approach is not determined within H but within some finite dimensional subspace S_h, where h is the discretization parameter. For $h \to 0$ we expect convergence to the solution of the continuous problem.

Theorem 4. *Given the energy functional*

$$J(v) := \frac{1}{2}a(v,v) - \langle f, v\rangle \to \min_{S_h}.$$

J(v) has its minimum in $u_h \in S_h$, if

$$a(u_h, \tilde{u}) = \langle f, \tilde{u}\rangle \ \forall \tilde{u} \in S_h \ , \tag{A.1}$$

where $a(u_h, \tilde{u})$ is again a positive bilinear form.
Proof: *For $u_h, \tilde{u} \in S_h$, $\epsilon \in \mathbb{R}$:*

$$\begin{aligned} J(u_h + \epsilon\tilde{u}) &= \frac{1}{2}a(u_h + \epsilon\tilde{u}, u_h + \epsilon\tilde{u}) - \langle f, u_h + \epsilon\tilde{u}\rangle \\ &= J(u_h) + \epsilon(a(u_h, \tilde{u}) - \langle f, \tilde{u}\rangle) + \frac{1}{2}\epsilon^2 a(\tilde{u}, \tilde{u}). \end{aligned} \tag{A.2}$$

If u_h fulfills equation (A.1), and $\epsilon = 1$:

$$J(u_h + \tilde{u}) = J(u_h) + \frac{1}{2}a(\tilde{u}, \tilde{u}) > J(u_h)$$

So, u_h is a unique minimum.
On the other hand, if J has a minimum at u_h, the derivative of the function $J(u_h + \epsilon\tilde{u})$ at $\epsilon = 0$ has to vanish for every $\tilde{u} \in V$. According to (A.2), this derivative is $a(u_h, \tilde{u}) - \langle f, \tilde{u}\rangle$, (A.1) follows.

Let $\{\phi_1, \phi_2, ..., \phi_N\}$ be a basis of S_h. Then, (A.1) is equivalent to

$$a(u_h, \phi_i) = \langle f, \phi_i\rangle, \ i = 1, 2, ..., N.$$

With

$$u_h = \sum_{k=1}^{N} x_k \phi_k$$

we get the linear system of equations

$$\sum_{j=1}^{N} a(\phi_j, \phi_i)x_j = \langle f, \phi_i\rangle, \ i = 1, 2, ..., N,$$

which can be rewritten in matrix-vector form

$$Ax = b \tag{A.3}$$

where $A_{ij} = a(\phi_j, \phi_i)$ and $b_i = \langle f, \phi_i \rangle$. Matrix A is positive definite if a is an H-elliptic bilinear form:

$$x^\top Ax = \sum_{i,j} x_i A_{ij} x_j = a(\sum_j x_j \phi_j, \sum_i x_i \phi_i) = a(u_h, u_h) \geq \alpha ||u_h||_m^2.$$

A is often called stiffness matrix or system matrix.
This method of solving the positive definite problem (A.1) is called Ritz-Galerkin method. In the following, we derive some properties of this method before we describe features of the actual FEM implementation (cf. sec. A.3).

Theorem 5. Stability: *Independent of the subspace S_h that we choose, the solution of (A.1) satisfies*

$$||u_h||_m \leq \alpha^{-1} ||f|| \ .$$

Proof: *Suppose u_h is a solution of (A.1). Let $v = u_h$:*

$$\alpha ||u_h||_m^2 \leq a(u_h, u_h) = \langle f, u_h \rangle \leq ||f|| ||u_h||_m.$$

Theorem 6. Céa's Lemma: *a is an H-elliptic bilinear form. u and u_h be solutions of a variational problem in H and $S_h \subset H$. Then,*

$$||u - u_h||_m \leq \frac{C}{\alpha} \inf_{v_h \in S_h} ||u - v_h||_m.$$

Proof: *(A.1) yields*

$$a(u, v) = \langle f, v \rangle \ \ \forall v \in H$$
$$a(u_h, v) = \langle f, v \rangle \ \ \forall v \in S_h$$

As $S_h \subset H$, subtraction yields

$$a(u - u_h, v) = 0 \ \ \forall v \in S_h. \tag{A.4}$$

(A.4) is often referred to as Galerkin orthogonality. With $v = v_h - u_h \in S_h$ ($v_h \in S_h$), this yields $a(u - u_h, v_h - u_h) = 0$, and thus

$$\begin{aligned} \alpha ||u - u_h||_m^2 &\leq a(u - u_h, u - u_h) = a(u - u_h, u - v_h) + a(u - u_h, v_h - u_h) \\ &\leq C ||u - u_h||_m ||u - v_h||_m. \end{aligned}$$

Céa's Lemma states that the error of any Galerkin approximation is only a constant factor (independent of h) higher than that of the best approximation of u in V.

A.3. Finite Elements for Elliptic Systems

One usually solves variational problems in so-called "Finite Element Spaces". The domain Ω is divided into a finite number of sections (*elements*) and one considers functions that are polynomials on these elements.

A.3.1. Features of FEM Spaces

Features of finite element spaces include:

- Decomposition of the domain: In 2D, the elements are usually triangles or squares; in 3D, the domain is divided into tetrahedra or cubes.
- Continuity and differentiability properties: Finite elements are C^k elements, if they are contained in $C^k(\Omega)$. In this section, we will restrict ourselves to the simplest elements: (bi-) linear elements of class $C^0(\Omega)$.
- Polynomial degree: In this section, we will restrict ourselves to polynomials of degree 1. They have the form
$$u(x, y) := ax + by + c \ .$$

A.3.2. Triangulation

First, one needs an admissible triangulation:

- Non-overlapping elements: The intersection of two elements T_i,T_j should be empty:
$$T_i \cap T_j = \{\}, i, j \in \{1, ..., t\}, i \neq j$$
- Coverage of the domain: The union of all t triangles is equal to the domain in which the problem is posed
$$\bigcup_{k=1}^{t} T_k = \Omega \ .$$

In this section, we will explain how to perform a finite element discretization on an irregular triangular grid (cf. 4.2.2). The computations on a regular (triangular) grid (cf. e.g. sec. 3.5) is just a specific (simpler) case of this procedure.

A.3.3. Set-up of the System Matrix

The system matrix can be set up elementwise:

$$a(u, v) = \int_\Omega \sum_{k,l} a_{kl} \partial_k u \partial_l v dx \ .$$

This yields

$$A_{ij} = a(\phi_i, \phi_j) = \int_\Omega \sum_{k,l} a_{kl} \partial_k \phi_i \partial_l \phi_j dx = \sum_{T \in \mathcal{T}} \int_T \sum_{k,l} a_{kl} \partial_k \phi_i \partial_l \phi_j dx \ . \qquad \text{(A.5)}$$

As we are considering elements with compact support, the sum only has to include those triangles that are contained in the support of ϕ_i *and* ϕ_j.
In the case of an irregular triangulation, we use element-oriented computations: For every element $T \in \mathcal{T}$ (where $\mathcal{T}$ is the set of all triangles), we compute the individual share to the corresponding system matrix entry.

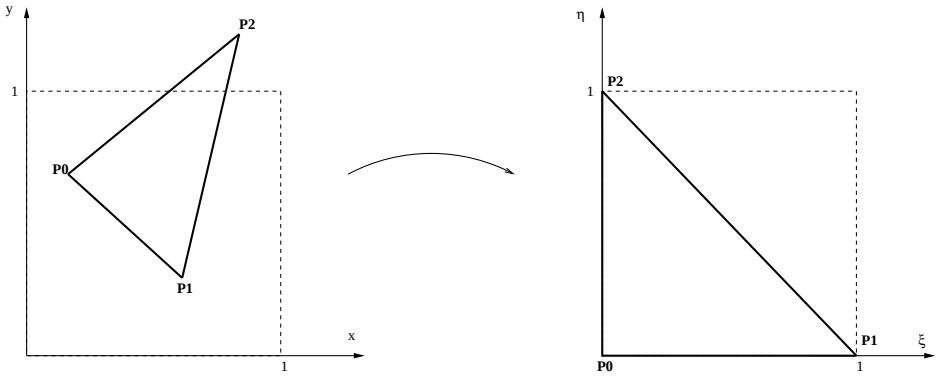

Figure A.1.: Transformation of an arbitrary triangle into a unit triangle

In order to simplify the occurring integrations of each triangle we transform the arbitrary triangles into unit triangles. Figure A.1 shows the linear transformation to a local coordinate system.

A general triangle with the vertices $P_0(x_0, y_0)$, $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ that is numbered counter-clockwise (cf. fig. A.1) can be transformed bijectively into an isosceles, orthogonal triangle T_0 with an edge length of one:

$$\begin{aligned} x &= x_0 + (x_1 - x_0)\xi + (x_2 - x_0)\eta \ , \\ y &= y_0 + (y_1 - y_0)\xi + (y_2 - y_0)\eta \ . \end{aligned} \tag{A.6}$$

With this substitution of variables, the computation of the integral over the triangle T is transformed into a simple area integral. Thus $dxdy$ is to be replaced by

$$dxdy = Jd\xi d\eta$$

where

$$J = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ P_0 & P_1 & P_2 \end{vmatrix} = (x_1 - x_0)(y_2 - y_0) - (x_2 - x_0)(y_1 - y_0)$$

is the so-called Jacobi determinante.
Integrating some function u over a general triangle then becomes:

$$\int_{T_{gen.}} u(x)dx = \int_{T_{unit}} u(x(\xi,\eta), y(\xi,\eta))Jd\xi d\eta = J\int_0^1 \int_0^{1-\eta} u(\xi)d\xi d\eta \ .$$

(A.5) therefore has the following entries:

$$A_{i,j} = \sum_{T\in\mathcal{T}} J\int_0^1 \int_0^{1-\eta} \sum_{k,l} a_{kl}\partial_k\phi_i\partial_l\phi_j d\xi d\eta \ .$$

As we want to use *linear* Finite Elements, the representation of the functions ϕ_i in local coordinates is:

$$\begin{aligned} \phi_{P0} &= 1 - \xi - \eta \ , \\ \phi_{P1} &= \xi \ , \\ \phi_{P2} &= \eta \ . \end{aligned}$$

The connection to global coordinates is given by equation (A.6) and can be written as the matrix relation

$$\begin{pmatrix} 1 \\ x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ P_0 & P_1 & P_2 \end{pmatrix} \begin{pmatrix} \phi_{P0} \\ \phi_{P1} \\ \phi_{P2} \end{pmatrix} \ .$$

Given the system matrix, the next step is to solve (A.3). If a is an h-elliptic bilinear form, we can solve this sparse positive definite system by some iterative solver. In the following two sections, we will describe two possible solvers that we used for the solving of our linear systems: the method of conjugate gradients (sec. (A.4.2)) and the multigrid method (sec. (A.5)).

A.4. Gradient Descent Methods

If one wants a large system of equations with a positive definite system matrix, one uses the fact that the solution of the equation $Ax = b$ is just the minimum of

$$f(x) = \frac{1}{2} x^\top A x - b^\top x \tag{A.7}$$

The simplest method that uses this fact is the *general gradient approach*.

A.4.1. The General Gradient Method

Starting at some initial guess x_0, one first computes the negative gradient of the function f at that point. For quadratic functions (A.7), this yields

$$d_0 = -\nabla f(x_0) = b - A x_0 \ .$$

Then, one computes the minimum of f on the line $\{x_0 + t d_0 : t \geq 0\}$. This minimum is at $t = \alpha$:

$$\alpha = \frac{d_0^\top d_0}{d_0^\top A d_0} \ .$$

One therefore has a new estimate $x_1 = x_0 + \alpha d_0$ and can (again) compute the negative gradient, and so on. It is clear that this iterative approach yields a series (x_k) with $f(x_0) \geq f(x_1) \geq f(x_2) \geq \ldots$. While it will always yield the global optimum (as long as the system is really positive definite), the general gradient method tends to converge very slowly for matrices with a high condition number [1].

A.4.2. Conjugate Gradient Method

This method was introduced by Hestenes and Stiefel [HS52]; it decreases the number of iterations (until convergence) considerably.
The conjugate gradient method computes a series of vectors p_i which are non-zero and satisfy $(p_i)^\top A p_j = 0$ for all $i \neq j$. This property is known as conjugacy. Starting again from an initial

[1] Unfortunately, all elliptic systems that are presented in this thesis generally have a high condition number.

x_0, the method computes a sequence x_k using conjugate vectors as bases. The iterative updates are

$$\begin{aligned} x_{k+1} &= x_k + \alpha_k p_k, \\ g_{k+1} &= g_k + \alpha_k A d_k, \\ d_{k+1} &= -g_{k+1} + \beta_k d_k, \text{ where} \\ \alpha_k &= \frac{g_k^\top g_k}{d_k^\top A d_k} \text{ and} \\ \beta_k &= \frac{g_{k+1}^\top g_{k+1}}{g_k^\top g_k} \end{aligned}$$

Preconditioned Conjugate Gradient Method

In many of our experiments, we use a variation of the classical conjugate gradient method: the so-called *preconditioned* conjugate gradient method (PCG) [CGO76]. When we use a linear translation $\hat{x} = B^{1/2}x$ for a non-singular matrix B, the system becomes

$$f(\hat{x}) = \frac{1}{2}\hat{x}^T(B^{-1/2}AB^{-1/2})\hat{x} - (B^{-1/2}\hat{x})^T b.$$

Then we apply the classical conjugate gradient method on this preconditioned matrix. The goal is to choose a matrix B so that the eigenvalues of $B^{-1/2}AB^{-1/2}$ are clustered closely to each other, thus reducing the number of iteration steps. The preconditioned CG algorithm is presented more precisely in algorithm 2. We confine ourselves to the use of the diagonal of A as B (as we use multigrid methods (cf. sec. A.5) if real-time performance is demanded). For a review of more advanced preconditioners, we refer to [Bra97].

Algorithm 2 Preconditioned CG Algorithm

Choose x_0. $g_0 = Ax_0 - b, d_0 = -h_0 = -B^{-1}g_0$.
For a given tolerance ϵ,

WHILE $(||g^k||/||b|| > \epsilon)$
$x_{k+1} = x_k + \alpha_k d_k$
$\alpha_k = \frac{g_k^\top h_k}{d_k^\top A d_k}$
$g_{k+1} = g_k + \alpha_k A d_k$
$h_{k+1} = B^{-1} + g_{k+1}$
$d_{k+1} = -h_{k+1} + \beta_k d_k$
$\beta_k = \frac{g_{k+1}^\top h_{k+1}}{g_k^\top h_k}$
$k = k + 1$;
END

A.5. Multigrid Methods

Multigrid methods are currently among the fastest available methods to solve elliptic PDEs. In fact, they are the only methods with a complexity of $O(n)$, where n is the linear system's number of unknowns. Multigrid methods were developed by Brandt [Bra77], and mathematically substantiated by Hackbusch [Hac85].
The main observation that led to the development of multigrid methods is the fact that standard solvers are *smoothing operators*: If we consider the error e (approximate solution $\tilde{x}$ of x in (A.3) minus true solution), we see that it consists mainly of low frequency components, while high frequency components are eliminated. It takes many additional iterations to eliminate the low frequency components as well.
This section is just to give an algorithmical introduction to multigrid methods – for a sound mathematical analysis, we refer to [Hac85, Hac93] and for a more detailed algorithmical introduction, we refer to [PTVF92]. The section is divided into three parts: Firstly, we sketch the standard *Gauss-Seidel* method, that is used as a smoother. Secondly, we outline how the so-called *coarse-grid correction* is performed. Finally, we summarize and illustrate the *full multigrid algorithm*.

A.5.1. Gauss-Seidel Method

If we order the mesh points from 1 to N, the Gauss-Seidel updating scheme is

$$x_i = -\frac{(\sum_{j=1, j\neq i}^{N} A_{ij}x_j - f_i)}{A_{ii}}, \quad i = 1, ..., N \ .$$

Note that the new values of x are used as soon as they become available. A simple enhancement of the traditional Gauss-Seidel Method is the *Red-Black Gauss-Seidel Method*: The individual entries of the system matrix are "colored" in a checkerboard fashion. One makes one sweep to update the red nodes and then a second sweep to update the black nodes.

A.5.2. Coarse-Grid Correction

We start with an initial estimate of $\tilde{x}_h$ and compute the so-called *defect*:

$$d_h = A_h\tilde{x}_h - f_h, \tag{A.8}$$

where h denotes the mesh size of some uniform grid. Since A_h is linear, the defect satisfies

$$A_h e_h = -d_h, \tag{A.9}$$

where e_h is the difference between estimated $\tilde{x}_h$ and true x_h. Let us now *coarsify* the defect onto a grid with mesh size $2h$:

$$d_{2h} = \mathcal{R}d_h,$$

where $\mathcal{R}$ is a restriction operator. As we use conforming finite elements, this operator is directly given. For details we refer to [Bra97]. Then, we can solve (A.9) for the error (correction) e; this operation, however, is performed on the coarse grid:

$$A_{2h}e_{2h} = -d_{2h} \ . \tag{A.10}$$

Finally, we interpolate[2] the error e_{2h} onto the finer grid e_h:

$$e_h = \mathcal{P} e_{2h}$$

and update the approximation $\tilde{x}_h$:

$$\tilde{x}_h^{new} = \tilde{x}_h + e_h \ . \tag{A.11}$$

It is clear that this procedure (in contrast to the Gauss-Seidel method from the last section) can reduce low-frequency errors. High-frequency errors, on the other hand, do not even have a representation on the fine grid and therefore won't be reduced. It therefore makes sense to combine the ideas of coarse-grid correction and Gauss-Seidel relaxation in the way described in algorithm 3:

Algorithm 3 Two-Grid Iteration

1. **Pre-Smoothing:** Perform n Gauss-Seidel sweeps (starting with some initial value) on grid h. This yields $\tilde{x}_h$ for eq. A.8.
2. **Coarse-grid correction:** Executing above algorithm (A.8) - (A.11) yields x_h^{new}.
3. **Post-Smoothing:** Perform m Gauss-Seidel sweeps starting with x_h^{new}.

A.5.3. Full Multigrid Algorithm

There are two additional ideas that yield the full multigrid algorithm:

1. Instead of solving (A.10) exactly on grid level $2h$, we use even coarser grids and recursively apply the two-grid algorithm 3. Usually, one does not need more than $\gamma = 2$ iterations of algorithm 3. Figure A.2 shows the details.
2. Additionally, we do not start with some initial guess, but with the exact solution given at the coarsest possible grid level (where the whole problem size is, e.g., only 3×3). This coarse solution is interpolated to the next grid level

   $$x_h = \mathcal{P} x_{2h}. \tag{A.12}$$

   Instead of the V or W cycles as seen in figure A.2, this yields to a series of increasingly tall N's (cf. fig. A.3).

[2] Again, the projection operator $\mathcal{P}$ is directly given if we use finite elements.

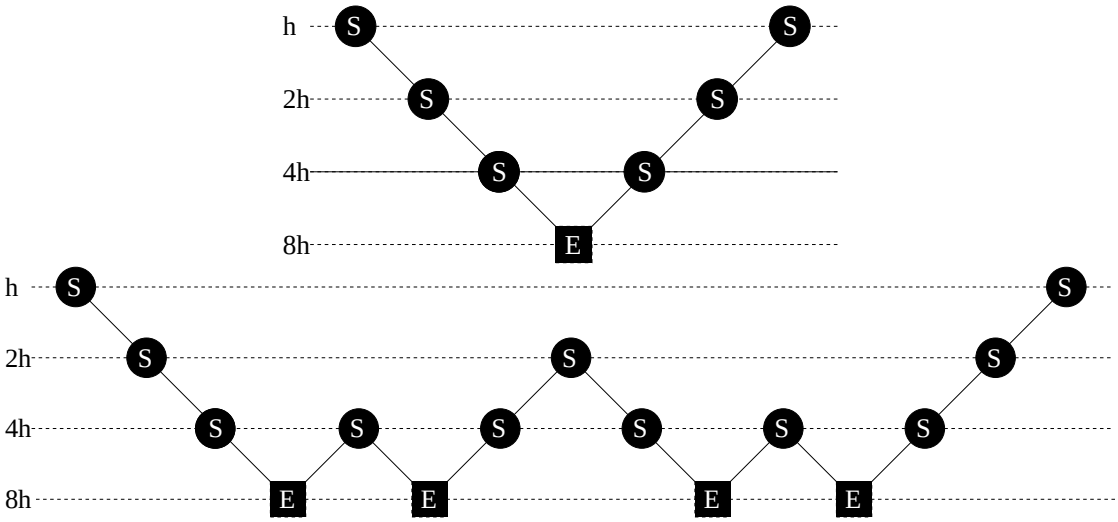

Figure A.2.: Multigrid algorithm with 4 grid levels. S means smoothing (relaxation), E denotes exact computation (on the coarsest level only), diagonal lines to the bottom ($\backslash$) denote restriction operators, diagonal lines to the top ($/$) note prolongation. **Top:** $\gamma = 1$. **Bottom:** $\gamma = 2$.

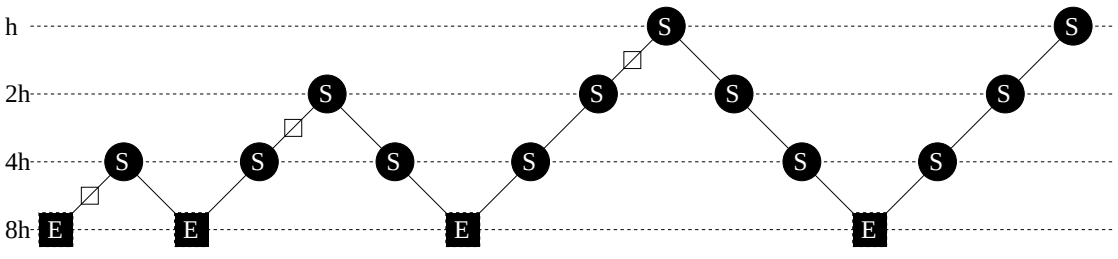

Figure A.3.: Multigrid algorithm with 4 grid levels. The symbol $\square$ means that a coarse-grid solution has to be prolongated with (A.12).

B. Saddle-Point Problems and Mixed FEM

We have seen in this thesis, that imposing additional constraints (like, e.g., incompressibility) yields quadratic problems with linear constraints (cf. e.g.(5.25),(5.36)):

$$\begin{aligned} J(u) &= \frac{1}{2}a(u,u) - \langle f, u\rangle \\ &\text{s.t. } b(u,\mu) = \langle g, \mu\rangle \ \forall \mu \in M \end{aligned} \tag{B.1}$$

that can be classified as *saddle point problems*. This chapter is to sketch basic mathematical properties of saddle-point problems (sec. B.1), features of the corresponding discretization with mixed finite elements (sec. B.2), and finally the solution of the discretized system using the so-called Uzawa algorithm (sec. B.3).

B.1. Saddle-Point Problems

The corresponding Lagrange functional to problem (B.1) is

$$\mathcal{L}(u,\lambda) := J(u) + (b(u,\lambda) - \langle g, \lambda\rangle).$$

Because $\mathcal{L}(u,\lambda)$ is a quadratic form in (u,λ), this yields

$$\begin{aligned} a(u,v) + b(v,\lambda) &= \langle f, v\rangle, \ \forall v \in X \\ b(u,\mu) &= \langle g, \mu\rangle, \ \forall \mu \in M, \end{aligned} \tag{B.2}$$

with the saddle point property

$$\mathcal{L}(u,\mu) \leq \mathcal{L}(u,\lambda) \leq \mathcal{L}(v,\lambda), \ \forall (v,\mu) \in X \times M$$

for every (u,λ) that solves (B.2).
X and M denote Hilbert spaces and X', M' the corresponding dual spaces. $a : X \times X \to \mathbb{R}$, and $b : X \times M \to \mathbb{R}$ be continuous bilinear forms. The problem of finding $(u,\lambda) \in X \times M$ with (B.2) defines the linear mapping

$$\begin{aligned} L : X \times M &\to X' \times M' \\ (u,\lambda) &\to (f,g) \ . \end{aligned}$$

Let us rewrite (B.2) as an operator equation. To do so we have to assign mappings to the bilinear forms $a(u,v)$, $b(u,\mu)$, and $b(v,\lambda)$:

$$\begin{aligned} A : X \to X', \ \langle Au, v\rangle &= a(u,v), \ \forall v \in X \\ B : X \to M', \ \langle Bu, \mu\rangle &= b(u,\mu), \ \forall \mu \in M \\ B^\top : X \to X', \ \langle B^\top \lambda, v\rangle &= b(v,\lambda) \ \forall v \in X \ , \end{aligned}$$

and (B.2) can be rewritten as

$$\begin{aligned} Au + B^\top \lambda &= f, \\ Bu &= g. \end{aligned} \tag{B.3}$$

Furthermore, we define the following spaces

$$\begin{aligned} V(g) &:= \{v \in X; b(v,\mu) = \langle g, \mu \rangle,\ \forall \mu \in M\}, \\ V &:= \{v \in X; b(v,\mu) = 0,\ \forall \mu \in M\}. \end{aligned} \tag{B.4}$$

Due to continuity of b, V is a closed subspace of X.

Theorem 7. *The saddle-point problem (B.3) has a unique solution $(u, \lambda) \in X \times M$, if*

1. *the bilinear form a is V-elliptic (with V according to (B.4)):*

$$a(v,v) \geq \alpha ||v||^2,\ \forall v \in V\ ,$$

   and

2. *the bilinear form b fulfills the so-called* inf-sup *(or* Babuška-Brezzi*) condition*

$$\inf_{\mu \in M} \sup_{v \in X} \frac{b(v,\mu)}{||v|| ||\mu||} \geq \beta > 0\ .$$

For a proof of theorem 7, we refer to [BF91].

B.2. Mixed FEM

Analogous to sec. A.2-A.3 for elliptic problems, we want to find an adequate discretization for saddle point problems. Again, we choose finite dimensional subspaces $X_h \subset X$, and $M_h \subset M$. The discretized version of (B.2) is

$$\begin{aligned} a(u_h, v) + b(v, \lambda_h) &= \langle f, v \rangle\ \forall v \in X_h \\ b(u_h, \mu) &= \langle g, \mu \rangle\ \forall \mu \in M_h. \end{aligned} \tag{B.5}$$

We try to find $(u_h, \lambda_h) \in X_h \times M_h$ that fulfill (B.5). This type of approach is called *mixed finite element method*.
Analogous to (B.4), we define

$$V_h := \{v \in X_h; b(v,\mu) = 0,\ \forall \mu \in M_h\}.$$

Note that the fact that $X_h \subset X$ does not necessarily mean that $V_h \subset V$. Therefore, a is not necessarily V_h-elliptic.

Definition 8. *A family of finite element spaces X_h, M_h fulfills the Babuška-Brezzi condition if there exist $\alpha > 0$ and $\beta > 0$ (independent of h) with the following properties:*

1. *The bilinear form a is V_h elliptic*

$$a(v_h, v_h) \geq \alpha ||v_h||^2,\ \forall v_h \in V_h$$

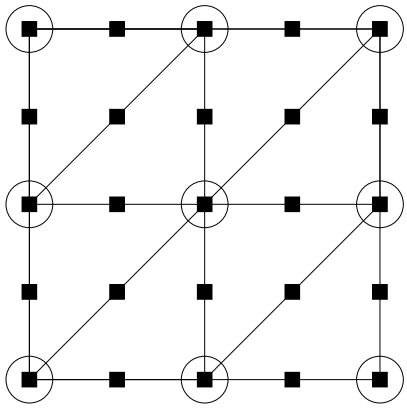

Figure B.1.: Sketch of 2D Taylor-Hood elements: Center position of linear elements for functions in M_h coincide with the triangle vertices (circles). Quadratic elements for functions in X_h are defined also on intermediate positions (squares).

2. *The so-called Brezzi condition is fulfilled:*

$$\sup_{v \in X_h} \frac{b(v, \lambda_h)}{||v||} \geq \beta ||\lambda_h||, \ \forall \lambda_h \in M_h \ .$$

Theorem 9. *X_h, M_h fulfill the Babuška-Brezzi condition. Then*

$$||u - u_h|| + ||\lambda - \lambda_h|| \geq c\{ \inf_{v_h \in X_h} ||u - v_h|| + \inf_{\mu_h \in M_h} ||\lambda - \mu_h||\}.$$

For a proof, we refer to [Bra97].
To summarize, the spaces X_h and M_h must be harmonized. A family of elements that is often used for Stokes (and Stokes-like) systems is the so-called *Taylor Hood element*:
For discretizing functions of the space X_h, we use polynomials of degree 2. If, e.g., linear triangles are used, this yields the interpolation function

$$u(x_1, x_2) := ax_1^2 + bx_2^2 + cx_1x_2 + dx_1 + ex_2 + f \ .$$

For functions of the space M_h, polynomials of degree 1 are used:

$$p(x_1, x_2) := ax_1 + bx_2 + c \ .$$

Figure B.1 shows the arrangement of the individual basis functions. For discretizing the Stokes equation (5.17a), its adjoint (5.17b), and the vorticity transport regularizer (5.35) we use a slight modification of the standard Taylor-Hood elements. Instead of a triangular mesh, we employ a regular quadratic mesh and define bilinear (for functions in M_h), and biquadratic (for functions in X_h) finite elements. A proof that these Taylor-Hood elements fulfill the Babuška-Brezzi condition can be found in [Ver84, GR86].

B.3. Uzawa Algorithm

Discretization of the saddle-point problem (B.5) in some appropriate finite element space yields the following linear system of equations:

$$\begin{aligned} Au + B^\top \lambda &= f \ , \\ Bu &= 0 \ , \end{aligned} \tag{B.6}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{m\times n}$, $u, f \in \mathbb{R}^n$, $\lambda, g \in \mathbb{R}^m$. The saddle point problems that we consider in this thesis contain a submatrix A that is positive definite. One can therefore solve the first equation of the system (B.6) for the unknown u:

$$u = A^{-1}(f - B^\top p)$$

and insert the result in the second equation

$$BA^{-1}(f - B^\top p) = 0 \ .$$

This gives a system which only incorporates the pressure

$$(BA^{-1}B^T)p = BA^{-1}f \ . \tag{B.7}$$

The matrix $(BA^{-1}B^\top)$ is symmetric and positive definite. Therefore, we apply the conjugate gradient algorithm (cf. sec. A.4.2) to (B.7). This requires a single matrix inversion in every iteration step. For computational efficiency, this is accomplished using a multigrid scheme (cf. sec. A.5). Algorithm 4 details the overall process.

Algorithm 4 Uzawa Algorithm

$p_0 \in \mathbb{R}$, $Au_1 = f - B^\top p_0$. Set $d_1 = -q_1 = Bu_1$.
repeat
  $p_k = B^\top d_k$
  Approx. $h_k = A^{-1}p_k$ using multigrid
  $\alpha_k = d_k^\top d_k / (p_k^\top h_k)$
  $p_k = p_{k-1} - \alpha_k d_k$
  $u_{k+1} = u_k + \alpha_k h_k$
  $q_{k+1} = -Bu_{k+1}$
  $\beta_k = q_{k+1}^\top q_{k+1} / (q_k^\top q_k)$
  $d_{k+1} = -q_{k+1} + \beta_k d_k$
until $||q_{k+1}|| < \epsilon$

C. The Equations of Fluid Motion

In this section we develop the fundamental equations of fluid mechanics. Note that we just give a short overview about the equations that we will need in chapter 5. For a more complete coverage of the topic, we refer to [CM93] and [And95]. Section C.1 deals with the equations that arise from the simplest assumptions. We will show that this description is not able to describe all flow effects and therefore introduce the full Navier-Stokes equations in section C.2. In section C.3, we derive a special form of the Navier-Stokes equations – the vorticity transport equation, which we use in sec. 5.4 as a physically motivated prior. After introducing and motivating the dimensionless Reynolds number in sec. C.4, we finally present the linearized form of the Navier-Stokes equations, the so-called Stokes equation, in sec. C.5. This equation is used as a prior in sec. 5.3.

C.1. Euler's Equation (Inviscid Flow)

We consider the flow of a fluid within a certain volume Ω using an Eulerian description with $u(x,t)$ representing the velocity of the fluid at x at time t.

The derivation of Euler's equation is based on three basic conservation laws:

(i) conservation of mass

(ii) conservation of momentum

(iii) energy conservation.

We will introduce these conservation laws in the following sections.

C.1.1. Conservation of Mass

Let $\rho(x,t)$ denote the density (i.e. mass of the fluid particles per unit of volume) of a fluid at (x,t). Let W be a fixed subregion of Ω with boundary ∂W. The total mass m of the fluid in W at time t is given by

$$m(W,t) = \int_W \rho(x,t)\, dV$$

where dV is the volume element.
The rate of change of mass in W is

$$\frac{d}{dt} m(W,t) = \frac{d}{dt} \int_W \rho(x,t)\, dV \tag{C.1}$$

$$= \int_W \frac{\partial \rho}{\partial t}(x,t)\, dV\ . \tag{C.2}$$

The mass of fluid at time t passing out through the boundary ∂W per unit of time is given by the surface integral of $\rho u \cdot n$ over ∂W, where n is the unit outward normal, defined at points of ∂W. Therefore, the rate of increase of mass in W equals the rate at which mass is crossing ∂W in the *inward* direction. This observation gives rise to the *integral form of the conservation law of mass*:

$$\int_W \frac{\partial \rho}{\partial t}(x,t)dV = -\int_{\partial W} \rho u \cdot n dA \; . \tag{C.3}$$

Using the divergence theorem

$$\int_W \frac{\partial \rho}{\partial t}(x,t)dV = -\int_W \nabla \cdot (\rho u)dV$$

we can rewrite (C.3), obtaining the *differential form of the law of conservation of mass*, also known as the *continuity equation*:

$$\int_W \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) = 0 \; .$$

C.1.2. Conservation of Momentum

The velocity field is given by

$$u(x(t),t) = \frac{dx}{dt}(t) \; .$$

The acceleration a of a fluid particle is therefore

$$a(t) = \frac{d^2}{dt^2}x(t) = \frac{d}{dt}u(x(t)) \; .$$

Using the chain rule, this is equal to

$$a(t) = \frac{\partial u}{\partial t} + (u \cdot \nabla)u.$$

For the derivation of Euler's equation, the fluid is considered an *ideal fluid*. This means that for any motion of the fluid, there is a function $p(x,t)$ (pressure) with the following property: If S is a surface in the fluid with a unit normal n, the force of stress (b) exerted across the surface S per unit area at $x \in S$ at time t is

$$b = p(x,t)n \; .$$

This means that the force acts orthogonally to the surface S, i.e. there are no tangential forces. Therefore, the concept of ideal fluids excludes many interesting real phenomena (a rotation can neither be started nor stopped)[1].
If W is a region in the fluid at a particular instant of time t, the total force exerted on the fluid inside W by means of stress on its boundary is

$$S_{\partial W} = -\int_{\partial W} pndA.$$

[1] This is why we will introduce the full Navier-Stokes equations, that are able to explain phenomena like this, in sec. C.2.

If e is any fixed vector in space, the divergence theorem gives

$$e \cdot S_{\partial W} = -\int_{\partial W} pe \cdot n dA = -\int_W \nabla \cdot (pe) dV = -\int_W (\nabla p) \cdot e dV.$$

Therefore,

$$S_{\partial W} = -\int_W \nabla p dV.$$

If f(x,t) denotes the given body force per unit mass, the total body force is

$$F = \int_W \rho f dV.$$

By Newton's Second Law (force = mass · acceleration), the *differential form of the law of balance of momentum for inviscid flows* arises:

$$\rho \frac{Du}{Dt} = -\nabla p + \rho f. \tag{C.4}$$

C.1.3. Energy Conservation

In the 2D case we have four (unknown) functions: $u = (u_1, u_2)^\top$, ρ and p. Up to this point we have only deduced three equations. Therefore, one more equation is needed to specify the fluid motion completely.[2]
This fourth equation origins from the conservation of energy. The total energy is composed of the kinetic energy (visible) and the internal energy (invisible). We will confine ourselves to flows where all energy is kinetic. It can be shown (e.g. [CM93]) that this restriction is equivalent to the assumption of incompressibility: Thus, the fourth equation that completes Euler's equation is

$$\nabla \cdot u = 0 \ .$$

For other types of energy equations (e.g. for isentropic fluids), we refer to [CM93].

C.2. Navier-Stokes Equation (Viscous Flow)

After the analysis of *ideal* fluids in the preceding section, we want to concentrate on the analysis of more general fluids: *viscous flow*. The continuity equation (conservation of mass) and energy conservation do not change, only the conservation of momentum is affected.
Now we have to take into account *shear stresses* and *normal stresses*. The shear stress τ_{xy} is related to the time rate of change of the shearing deformation of the fluid element, whereas the normal stress τ_{xx}, is related to the time rate of change of volume of the fluid element. As a result, both shear and normal stresses depend on velocity gradients in the flow.
Let us only consider the *surface* forces in x direction. These are:

- Net pressure p:

$$\big(p - (p + \frac{\partial p}{\partial x_1} dx_1)\big) dx_2 \ ,$$

[2] In 3D, we have a similar problem: four equations and five unknown functions.

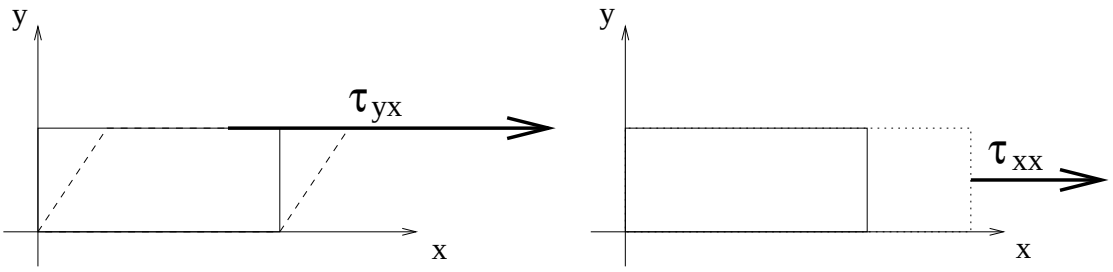

Figure C.1.: **Left:** shear stress **Right:** normal stress

- Net normal stress:
$$\big((\tau_{x_1x_1} + \frac{\partial \tau_{x_1x_1}}{\partial x_1} dx_1) - \tau_{x_1x_1}\big) dx_2 \ ,$$
- Net shear stress (shear towards x_2 direction):
$$\big((\tau_{x_2x_1} + \frac{\partial \tau_{x_2x_1}}{\partial x_2} dx_2) - \tau_{x_2x_1}\big) dx_1 \ .$$

Here, dx_1 and dx_2 denote the size of the 2D fluid elements in the specific dimensions. Adding all these surface force terms and the body force term yields the total force F_{x+1} in x_1 direction:

$$F_{x_1} = \big(- \frac{\partial p}{\partial x_1} + \frac{\partial \tau_{x_1x_1}}{\partial x_1} + \frac{\partial \tau_{x_2x_1}}{\partial x_2}\big) dx_1 dx_2 + \rho f_{x_1} dx_1 dx_2$$

Therefore, we obtain for the x_1 component of the momentum equation for a viscous fluid

$$\rho \frac{Du_1}{Dt} = -\frac{\partial p}{\partial x_1} + \frac{\partial \tau_{x_1x_1}}{\partial x_1} + \frac{\partial \tau_{x_2x_1}}{\partial x_2} + \rho f_{x_1} \ , \tag{C.5}$$

and for the x_2 direction analogously

$$\rho \frac{Du_2}{Dt} = -\frac{\partial p}{\partial x_2} + \frac{\partial \tau_{x_1x_2}}{\partial x_1} + \frac{\partial \tau_{x_2x_2}}{\partial x_2} + \rho f_{x_2} \ . \tag{C.6}$$

Equations (C.5) and(C.6) are the Navier-Stokes equations that describe general viscous fluids.

Newton states that shear stress in a fluid is proportional to the time rate of strain, i.e., velocity gradients. Such fluids are called *Newtonian* fluids. The following equations hold true for Newtonian fluids [3]:

$$\begin{aligned}
\tau_{x_1x_1} &= \lambda(\nabla \cdot u) + 2\mu \frac{\partial u_1}{\partial x_1} \ , \\
\tau_{x_2x_2} &= \lambda(\nabla \cdot u) + 2\mu \frac{\partial u_2}{\partial x_2} \ , \\
\tau_{x_1x_2} &= \tau_{x_2x_1} = \mu\big(\frac{\partial u_2}{\partial x_1} + \frac{\partial u_1}{\partial x_2}\big) \ ,
\end{aligned}$$

[3] Nearly all fluids are Newtonian fluids – exception: blood flow.

where μ is the molecular viscosity coefficient and λ is the second viscosity coefficient.

Substitution of these equations into (C.5) and (C.6) and rewriting the equations as one vector equation yields

$$\rho \frac{Du}{Dt} = -\nabla p + (\lambda + \mu)\nabla(\nabla \cdot u) + \mu \Delta u + \rho f \ . \tag{C.7}$$

Equation (C.4) can therefore be replaced by eq. (C.7) in the viscous case.
In the incompressible case ($\rho = \rho_0 = const.$), the continuity equation becomes

$$\nabla \cdot u = 0 \tag{C.8}$$

and thus, the Navier-Stokes equations can be simplified to

$$\rho \frac{Du}{Dt} = -\nabla p + \mu \Delta u + \rho f \ . \tag{C.9}$$

Equations (C.8) and (C.9) are self-contained; there are three equations for the three dependent variables u_1,u_2 and p.

C.3. The Vorticity Transport Equation

Let $u = (u_1, u_2)^\top, u = u(x,t), x = (x_1(t), x_2(t))$, denote a two-dimensional velocity field. We have seen that, in the incompressible case, the momentum equation of the Navier-Stokes equations becomes

$$\frac{\partial u}{\partial t} + (u \cdot \nabla)u = -\nabla p' + \nu \Delta u \ , \tag{C.10}$$

where ν is the coefficient of kinematic viscosity and $p' = p/\rho_0$. When we apply $\nabla\times$[4] to the Navier-Stokes equations we get

$$\nabla \times \frac{\partial u}{\partial t} + \nabla \times (u \cdot \nabla)u = -\nabla \times \nabla p' + \nu \nabla \times \Delta u. \tag{C.11}$$

Recalling that we are considering incompressible fluids (i.e. $\nabla \cdot u = 0$), we can simplify (C.11) and have

$$\frac{\partial}{\partial t}(\nabla \times u) + (u \cdot \nabla)(\nabla \times u) = \nu \Delta (\nabla \times u). \tag{C.12}$$

Setting $\omega = \nabla \times u$, (C.12) becomes

$$\frac{D\omega}{Dt} = \frac{\partial}{\partial t}\omega + u \cdot \nabla \omega = \nu \Delta \omega \ .$$

This equation is known as the (incompressible) vorticity transport equation. Together with the boundary condition at $t = 0 : \ \omega(x,0) = \omega_0$, it describes the evolution of vorticity over time. Note that, in the absence of external forces that act on the fluid and together with the incompressibility condition (C.8) , this equation is capable of completely describing the flow.

[4] In the forthcoming sections, we will consider 2D flows only. We therefore define $\nabla \times u = \frac{\partial u_2}{\partial x_1} - \frac{\partial u_1}{\partial x_2}$.

C.4. The Reynolds Number

In this section, we discuss some scaling properties of the Navier-Stokes equations. We will introduce the famous Reynolds number that measures the effect of viscosity on the flow.

For some given problem, let U be a *characteristic velocity*, L be a *characteristic length* and T be a *characteristic time*. The choice of these numbers is more or less arbitrary: The characteristic velocity is usually the fluid velocity at "infinity", while the characteristic length is determined by the experimental setup (e.g. in the flow around a cylinder, it is usually the diameter of the cylinder). Note that the choice of U and L determines the choice of T ($T = L/U$). Let us measure the spatial coordinate u, the velocity x and time t as fractions of the quantities U,L and T:

$$x' = x/L, \qquad u' = u/U, \qquad ,t' = t/T,$$

where x', u' and t' are dimensionless numbers. Then, the (incompressible) Navier-Stokes equations (C.10) can easily be rewritten as:

$$\begin{aligned} \frac{\partial u'}{\partial t'} + (u' \cdot \nabla')u' &= -\nabla p' + \frac{\nu}{LU}\Delta' u', \\ \nabla' \cdot u' &= 0, \end{aligned} \tag{C.13}$$

where $p' = p/(\rho_0 U^2)$. (C.13) are the Navier-Stokes equations in dimensionless numbers. The Reynolds number Re is just

$$Re = \frac{LU}{\nu}.$$

C.5. Linearization: Stokes Equation

Let us revisit the dimensionless Navier-Stokes equations that we have introduced in the last section [5]:

$$\frac{\partial u}{\partial t} + (u \cdot \nabla)u = -\nabla p' + \frac{1}{Re}\Delta u + f\ . \tag{C.14}$$

This equation is built up by the

$$\textit{diffusion}\text{ term:}\quad \frac{1}{Re}\Delta u$$

and the

$$\textit{convective}\text{ term:}\quad (u \cdot \nabla)u\ .$$

To summarize: u is convected subject to pressure forces and body forces f, and, at the same time, it diffuses. In specific cases where we know that the Reynolds number is small (i.e. slow velocity, large viscosity, or small bodies) the *diffusion* term will be the dominant term in (C.14). In these cases, we can neglect the non-linear term in (C.14) and still get a good approximation to the solution of the Navier-Stokes equations:

$$\frac{\partial u}{\partial t} = -\nabla p' + \frac{1}{Re}\Delta u + f\ . \tag{C.15}$$

[5] For the sake of readability, we have simplified the variable names.

Equation (C.15) is called unsteady Stokes equation. If we additionally assume stationarity of the flow, we get

$$-\frac{1}{Re}\Delta u + \nabla p' = f \ .$$

This equation is called steady Stokes equation, and we use it as physically motivated prior knowledge in section 5.3.

D. Discretization of the Vorticity Transport Equation

This section shows our discretization of the vorticity transport equation

$$\frac{\partial \omega}{\partial t} + (u \cdot \nabla)\omega = \nu \Delta \omega. \tag{D.1}$$

The vorticity transport equation (D.1) is a parabolic equation, whose hyperbolic part is the advection equation

$$\frac{\partial \omega}{\partial t} + (u \cdot \nabla)\omega = 0$$

and whose elliptic part is the diffusion equation

$$\frac{\partial \omega}{\partial t} = \nu \Delta \omega.$$

In the following we will use a Crank-Nicholson second order temporal differencing scheme for solving the diffusion part, and a second order Fromm-van Leer scheme for solving the advection part of equation (D.1).

The domain Ω is divided into cells. The vorticity ω is defined in the centers of the individual cells: ω_{ij}^n is the vorticity at $x_1 = i, x_2 = j$, and $t = n$. Furthermore, we will need to define quantities on the cell boundaries: $\omega_{i+1/2,j}^n$ denotes, e.g., the vorticity on the boundary between cells (i, j) and $(i + 1, j)$.

D.1. Crank-Nicholson Scheme

Let us now discretize (D.1). We combine an explicit Fromm-van Leer scheme and an implicit Crank-Nicholson scheme:

$$\frac{\omega_{i,j}^{n+1} - \omega_{i,j}^n}{\Delta t} = -(u \cdot \nabla)\omega_{i,j}^{n+\frac{1}{2}} + \frac{\nu}{2}\Delta(\omega^n + \omega^{n+1})_{i,j} \tag{D.2}$$

In order to be independent of time step $t = n + 1$ (i.e. we want to define an explicit scheme), we define a time step $t = n + 1/2$ for the advective part. For the diffusion part, on the contrary, we will use an implicit scheme – therefore, we are allowed to define ω at the time step $t = n + 1$.

In order to compute the unknown $\omega_{i,j}^{n+1}$, we rewrite (D.2) as

$$\left(-\frac{\Delta t}{2}\nu\Delta \right) \omega_{i,j}^{n+1} = \omega_{i,j}^n - \Delta t(u \cdot \nabla)\omega_{i,j}^{n+\frac{1}{2}} + \frac{\Delta t}{2}\nu\Delta\omega_{i,j}^n. \tag{D.3}$$

After discretization, this yields the following system of equations:

$$L\omega = b$$

with

$$L = \left(\quad - \frac{\Delta t}{2} \nu \Delta \right).$$

The right-hand-side of equation (D.3) is therefore

$$b = \omega_{i,j}^{n} - \Delta t (u \cdot \nabla) \omega_{i,j}^{n+\frac{1}{2}} + \frac{\Delta t}{2} \nu \Delta \omega_{i,j}^{n} \tag{D.4}$$

that can be conveniently solved by some corresponding elliptic solver (cf. appendix A). The operator L is discretized using standard finite differences:

$$(L\omega)_{i,j} = (1 + 4\Delta t \nu)\, \omega_{i,j} - \Delta t \nu \left(\omega_{i+1,j} + \omega_{i-1,j} + \omega_{i,j+1} + \omega_{i,j-1} \right).$$

The following sections will concentrate on the discretization of (D.4) using a Fromm-van Leer scheme.

D.2. Fromm Scheme

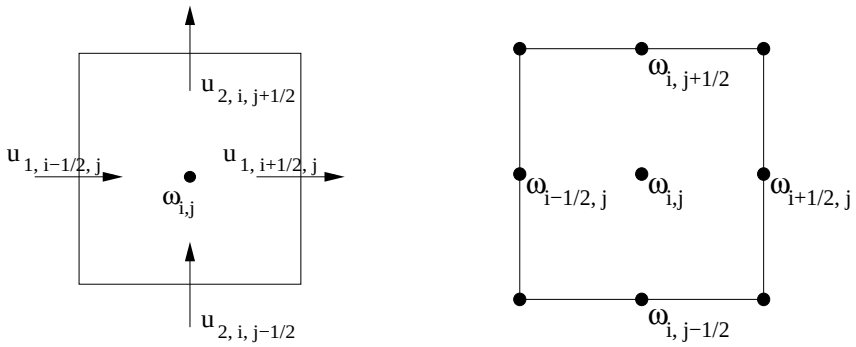

Figure D.1.: Finite Differences using Fromm Scheme. **Left:** Velocity components u_1 and u_2 on the cell edges. **Right:** Vorticity ω on the cell edges.

When discretizing (D.4), the main problem is the discretization of $(u \cdot \nabla)\omega_{i,j}^{n+1/2}$. As we are considering the *incompressible* vorticity transport equation, it is clear that

$$(u \cdot \nabla)\omega = \nabla \cdot (u\omega).$$

Using standard upwind-schemes for finite differences, $\nabla \cdot (u\omega)$ can be discretized as

$$\nabla \cdot (u\omega)_{i,j}^{n+\frac{1}{2}} = u_{1,i+\frac{1}{2},j} \omega_{i+\frac{1}{2},j}^{n+\frac{1}{2}} - u_{1,i-\frac{1}{2},j} \omega_{i-\frac{1}{2},j}^{n+\frac{1}{2}} + u_{2,i,j+\frac{1}{2}}^{n+\frac{1}{2}} \omega_{i,j+\frac{1}{2}} - u_{2,i,j-\frac{1}{2}} \omega_{i,j-\frac{1}{2}}^{n+\frac{1}{2}}.$$

The question remains, how to compute the vorticities at the four cell boundaries. We will only derive the formula for estimating the vorticity $\omega_{i+\frac{1}{2},j}^{n+\frac{1}{2}}$, the derivation of the other three vorticities is analogous. A linearized Taylor series for this right edge yields the following approximation:

$$\omega_{i+\frac{1}{2},j}^{n+\frac{1}{2}} \quad \approx \quad \omega_{i,j}^{n} + \frac{1}{2} \Delta x \frac{\partial \omega}{\partial x} + \frac{1}{2} \Delta t \frac{\partial \omega}{\partial t}. \tag{D.5}$$

For $\frac{\partial \omega}{\partial t}$, we can substitute our initial partial differential equation (D.1) and obtain

$$\begin{aligned} \omega_{i+\frac{1}{2},j}^{n+\frac{1}{2}} &= \omega_{i,j}^{n} + \frac{1}{2}\frac{\partial \omega}{\partial x_1} + \frac{1}{2}\Delta t \left(-u_1 \frac{\partial \omega}{\partial x_1} - \omega \frac{\partial u_1}{\partial x_1} - \frac{\partial (u_2 \omega)}{\partial x_2} \right) \\ &= \omega_{i,j}^{n} + \frac{1}{2}\left(1 - u_1 \Delta t\right) \frac{\partial \omega}{\partial x_1} - \frac{1}{2}\Delta t\, \omega\, \frac{\partial u_1}{\partial x_1} - \frac{1}{2}\Delta t \frac{\partial (u_2 \omega)}{\partial x_2} \,. \end{aligned} \tag{D.6}$$

The discretization of this equation is:

$$\begin{aligned} \omega_{i+\frac{1}{2},j}^{n+\frac{1}{2}} &= \omega_{i,j}^{n} + \frac{1}{2}\left(1 - u_{1,i+\frac{1}{2},j}\Delta t\right) \Delta_{x_1}^{VL} \omega_{i,j} - \frac{1}{2}\Delta t\, \omega_{i,j}^{n} \left(u_{1,i+\frac{1}{2},j} - u_{1,i-\frac{1}{2},j} \right) \\ &\quad - \frac{1}{2}\Delta t \left(u_{2,i,j+\frac{1}{2}} \omega_{i,j+\frac{1}{2}}^{upwind} - u_{2,i,j-\frac{1}{2}} \omega_{i,j-\frac{1}{2}}^{upwind} \right) \end{aligned} \tag{D.7}$$

where $\Delta_{x_1}^{VL}$ is the so-called van Leer slope, that will be defined in the next section. The term that introduces the u_2 component ($\frac{\partial (u_2 \omega)}{\partial x_2}$) is discretized using an *upwind* scheme: We think of the direction of information transfer as flowing from the upstream to downstream locations. Thus, to evaluate some quantity at a certain point, we only need information from the upstream region. For $\omega_{i,j+\frac{1}{2}}^{upwind}$ this yields, e.g.,

$$\omega_{i,j+\frac{1}{2}}^{upwind} = \begin{cases} \omega_{i,j}^{n} & \text{if} \quad u_{2,i,j+\frac{1}{2}} > 0 \\ \omega_{i,j+1}^{n} & \text{if} \quad u_{2,i,j+\frac{1}{2}} < 0 \end{cases} .$$

D.3. The van Leer Slope

The Fromm Scheme introduced in the preceding section is second-order accurate (we do not only have values for ω at the cell centers, but also values at the cell boundaries). However, if there are discontinuities in the flow, Fromm's second-order scheme will lead to oscillations that yield instabilities (cf. e.g. [Hir84]). A possibility to overcome this problem (but still retain higher-order accuracy) is the strategy to use Fromm's Scheme in smooth regions, but go back to first-order accuracy in regions where discontinuities are detected. This can be done by defining the so-called van Leer slope (cf. (D.7)):

$$\Delta_{x_1}^{VL} \omega_{i,j} = \begin{cases} sign(\omega_{i+1,j} - \omega_{i-1,j}) \min\{\text{slope}_{x_1}\} & \text{if} \quad \varphi_{x_1,i,j} > 0 \\ 0 & \text{if} \quad \varphi_{x_1,i,j} < 0 \end{cases} \tag{D.8}$$

where

$$\varphi_{x,i,j} = (\omega_{i+1,j} - \omega_{i,j})(\omega_{i,j} - \omega_{i-1,j}) \tag{D.9}$$

and

$$\text{slope}_{x_1} = \{2|\omega_{i,j} - \omega_{i-1,j}|, \frac{1}{2}|\omega_{i+1,j} - \omega_{i-1,j}|, 2|\omega_{i,j} - \omega_{i+1,j}|\}.$$

$\Delta_{x_2}^{VL}$ can be computed analogously.

Bibliography

[AB91] L. Amodei and M.N. Benbourhim. A vector spline approximation. *J. Approx. Theory*, 67(1):51–79, 1991.

[AD05] Y.D. Afanasyev and E.K. Demirov. A variational filtration and interpolation technique for piv employing fluid dynamical constraints. *Exp. Fluids*, 39(5):828–835, 2005.

[And95] J.D. Anderson. *Computational Fluid Dynamics. The Basics with Applications*. McGraw-Hill, 1995.

[Aur91] F. Aurenhammer. Voronoi diagrams – a survey of a fundamental geometric data structure. *ACM Computing Surveys*, 23(3):345–405, 1991. Habilitationsschrift. [Report B 90-09, FU Berlin, Germany, 1990].

[BAP94] D.H. Barnhart, R.J. Adrian, and G.C. Papen. Phase-conjugate holographic system for high-resolution particle image velocimetry. *Appl. Optics*, 33:7159–7170, 1994.

[BB95] S. S. Beauchemin and J. L. Barron. The computation of optical flow. *ACM Comput. Surv.*, 27(3):433–466, 1995.

[BCG96] C. Ballester, V. Caselles, and M. Gonzalez. Affine invariant segmentation by variational method. *SIAM J. Appl. Math.*, 56(1):294–325, 1996.

[BF91] F. Brezzi and M. Fortin. *Mixed and hybrid finite element methods*. Springer-Verlag New York, Inc., New York, NY, USA, 1991.

[BFB94] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 39:43–77, 1994.

[BG96] C. Ballester and M. Gonzalez. Texture segmentation by variational methods. In *ICAOS '96. 12th International Conference on Analysis and Optimization of Systems. Images, Wavelets and PDEs*, pages 187–193, 1996.

[BHY00] D. Béréziat, I. Herlin, and L. Younes. A generalized optical flow constraint and its physical interpretation. In *CVPR*, pages 2487–2492, 2000.

[BIK02] A. Borzi, K. Ito, and K. Kunisch. Optimal control formulation for determining optical flow. *SIAM J. Sci. Comput.*, 24(3):818–847, 2002.

[BL96] S. J. Baek and S. J. Lee. A new two-frame particle tracking algorithm using match probability. *Exp. Fluids*, 22(1):23–32, 1996.

[BLW03] M. Brede, A. Leder, and C. H. Westergaard. Time-resolved piv investigation of the separated shear layer in the transitional cylinder wake. In *Proc. 5th Int. Symp. on Particle Image Velocimetry (PIV'03),Busan, Korea*, 2003.

[BMT05] M.F. Beg, M. Miller, and A. Trouvé. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision*, 61(2):139–157, 2005.

[Bra77] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31(138):333–390, 1977.

[Bra97] D. Braess. *Finite elements. Theory, fast solver, and applications in solid mechanics*. Springer, 1997.

[Brü95] C. Brücker. Digital particle image velocimetry in a scanning light-sheet: 3d starting flow around a short cylinder. *Exp. Fluids*, 19:255–263, 1995.

[Brü96] C. Brücker. 3d scanning particle image velocimetry: Technique and application to a spherical cap wake flow. *Applied Scientific Research*, 56:157–179, 1996.

[Bur] S. Burgmann. Cylinder Wake Image Sequence provided by RWTH Aachen within the DFG Schwerpunktprogramm 1147, personal communication.

[BWF+05] A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnörr. Variational optic flow computation in real-time. *IEEE Trans. Image Proc.*, 14(5):608–615, 2005.

[CC70] S. Chapman and T.G. Cowling. *The Mathematical Theory of Non-Uniform Gases*. Cambridge University Press, 1970.

[CGO76] P. Concus, G.H. Golub, and D. P. O'Leary. A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations. *Academic Press, New York*, pages 309–332, 1976.

[CH] J. Carlier and D. Heitz. 2D turbulence sequence provided by Cemagref within the European Project 'Fluid Image Analysis and Description', personal communication.

[CHA+05] Th. Corpetti, D. Heitz, G. Arroyo, E. Mémin, and A. Santa-Cruz. Fluid experimental flow estimation based on an optical-flow scheme. *Exp. Fluids*, 40(1):80–97, 2005.

[Che03] D. Chetverikov. Applying feature tracking to piv. *International Journal of Pattern Recognition and Artificial Intelligence*, 17(4):477–504, 2003.

[Cia78] P.G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland Publ. Comp., 1978.

[CK05] J. Chen and J. Katz. Elimination of peak-locking error in piv analysis using the correlation mapping method. *Meas. Sci. Technol.*, 16:1605–1618, 2005.

[CM93] A. J. Chorin and J. E. Marsden. *A Mathematical Introduction to Fluid Mechanics*. Springer-Verlag, New York NY, 1993.

[CMP02] Th. Corpetti, E. Mémin, and P. Pérez. Dense estimation of fluid flows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(3):365–380, 2002.

[Coh96] L. D. Cohen. Auxiliary variables and two-step iterative algorithms in computer vision problems. *J. Math. Imaging Vis.*, 6(1):59–83, 1996.

[CV01] T. Chan and L. Vese. Active contours without edges, 2001.

[DHS04] D. H. Doh, T. G. Hwang, and T. Saga. 3d-ptv measurements of the wake of a sphere. *Meas. Sci. Technol.*, 15:1059–1066, 2004.

[ET99] T. Etoh and K. Takehara. A study on particle identification in particle image velocimetry - particle mask correlation method. *Journal of Visualization*, 1(3), 1999.

[FG83] M Fortin and R. Glowinski. *Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-value Problems*. North-Holland, Amsterdam, 1983.

[GM98] U. Grenander and M. Miller. Computational anatomy: An emerging discipline. *Quart. Appl. Math.*, 56(4):617–694, 1998.

[GP96] S. Gupta and J. Prince. Stochastic models for div-curl optical flow methods. *Signal Proc. Letters*, 3(2):32–34, 1996.

[GR86] V. Girault and P.-A. Raviart. Finite element methods for navier-stokes equations. *Springer*, 1986.

[Gun02] M. Gunzburger. *Perspectives in Flow Control and Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2002.

[GW87] R. C. Gonzales and P. Wintz. *Digital image processing*. Addison-Wesley Longman Publishing Co.,Inc, Boston, MA, USA, 1987.

[Hac85] W. Hackbusch. *Multigrid Methods and Applications*. Springer, 1985.

[Hac93] W. Hackbusch. *Iterative Solution of Large Sparse Systems of Equations*, volume 95 of *Applied Mathematical Sciences*. Springer, 1993.

[HC91] Y. Hassan and R. Canaan. Full-field bubbly flow velocity measurements using a multiframe particle trackingg technique. *Exp. Fluids*, 12:49–60, 1991.

[HF01] H.W. Haussecker and D.J. Fleet. Computing optical flow with physical models of brightness variation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(6):661–673, 2001.

[HFW93] H.T. Huang, H.F. Fielder, and J.J. Wang. Limitation and improvement of piv, ii. particle image distortion, a novel technique. *Exp. Fluids*, 15:263–273, 1993.

[HGSJ99] H.W. Haussecker, C.S. Garbe, H. Spies, and B. Jähne. A total least squares framework for low-level analysis of dynamic scenes and processes. In *Pattern Recognition, Proc. 21th DAGM Symposium*, pages 240–249, 1999.

[Hir84] C. Hirsch. *Numerical Computation of Internal and External Flows. Volume 2*. Wiley-Interscience, 1984.

[HK05a] R. Hain and C.J. Kähler. Advanced evaluation of time-resolved piv image sequences. In *Proc. 6th International Symposium on Particle Image Velocimetry, Pasadena, California, USA, September 21-23*, 2005.

[HK05b] R. Hain and C.J. Kähler. Fundamentals in multiframe piv. *Exp. Fluids*, 2005. Submitted.

[HNC+06] D. Heitz, V. Navaza, J. Carlier, T. Corpetti, and E. Mémin. Weighting the prior of a fluid dedicated optical flow estimator. In *13th International Symposium on Applications of Laser Techniques to Fluid Mechanics, Lisbon, Portugal*, 2006.

[HS52] M. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bur. Stand.*, 49:409–436, 1952.

[HS81] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[HS97a] J. M. Hyman and M. J. Shashkov. Adjoint operators for the natural discretizations of the divergence, gradient and curl on logically rectangular grids. *Appl. Numer. Math.*, 25(4):413–442, 1997.

[HS97b] J. M. Hyman and M. J. Shashkov. Natural discretizations for the divergence, gradient, and curl on logically rectangular grids. *Comput. Math. Appl.*, 33(4):81–104, 1997.

[HS99] J. M. Hyman and M. J. Shashkov. The orthogonal decomposition theorems for mimetic finite difference methods. *SIAM J. Numer. Anal.*, 36(3):788–818 (electronic), 1999.

[Jäh97] B. Jähne. *Digitale Bildverarbeitung*. Springer, Berlin, 1997.

[JCT01] M.C. Jullien, P. Castiglione, and P. Tabeling. Intermittency of a passive tracer in the inverse energy cascade. *Physical Review*, E 64(3), 2001.

[JJDAF95] K. Jambunathan, X.Y. Ju, B.N. Dobbins, and S. Ashforth-Frost. An improved cross correlation technique for particle images velocimetry. *Meas. Sci. Technol.*, 6:507–514, 1995.

[KLM94] G. Koepfler, C. Lopez, and J. M. Morel. A multiscale algorithm for image segmentation by variational method. *SIAM Journal on Numerical Analysis*, 31(1):282–299, 1994.

[KMK05] Y.-H. Kim, A.M. Martinez, and A.C. Kak. Robust motion estimation under varying illumination. *Image and Vision Computing*, 23(4), 2005.

[KMS03] T. Kohlberger, E. Mémin, and C. Schnörr. Variational dense motion estimation using the helmholtz decomposition. In L.D. Griffin and M. Lillholm, editors, *Scale Space Methods in Computer Vision*, volume 2695 of *LNCS*, pages 432–448. Springer, 2003.

[KMW96] R. Krishnamurthy, P. Moulin, and J. Woods. Multiscale motion models for scalable video coding. In *Proc. IEEE Int. Conf. Image Processing, Lausanne, Switzerland*, pages 965–968, 1996.

[KSBW04] T. Kohlberger, C. Schnörr, A. Bruhn, and J. Weickert. Parallel variational motion estimation by domain decomposition and cluster computing. In T. Pajdla and J. Matas, editors, *Proc. ECCV 2004*, volume 3024, pages 205–216. Springer, 2004.

[KSS89] T. Kobayashi, T. Saga, and S. Segawa. Multipoint velocity measurement for unsteady flow field by digital image processing. *Journal of Visualization*, 5:197–202, 1989.

[KYKI98] A. Kaga, K Yamaguchi, A Kondo, and Y. Inoue. Combination of piv data with cfd using cost function method. In *Proc. 8th Int. Symp. on Flow Visualization*, page 257, 1998.

[LaV01] LaVision. *FlowMaster: Advanced PIV Systems for Quantitative Flow Field Analysis*, 2001.

[LaV05] LaVision. *DaVis. Software for Intelligent Imaging.*, 2005.

[Lio71] J.L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer, 1971.

[LK81] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision, 1981.

[LL52] L. D. Landau and E. M. Lifschitz. *Hydrodynamik*. Number 6 in Lehrbuch der theoretischen Physik. Verlag Harri Deutsch, Thun, 5. (1991) edition, 1952.

[Lu96] H.Z. Lu. Simulation des écoulements externes visqueux incompressibles par une méthode de couplage: Différences finies particulaires. Technical report, LIMSI, 1996.

[Maa92a] H.-G. Maas. Complexity analysis for the establishment of image correspondences of dense spatial target fields. *International Archives of Photogrammetry and Remote Sensing*, 29(B5):102–107, 1992.

[Maa92b] H.-G. Maas. Digitale photogrammetrie in der dreidimensionalen strömungsmesstechnik. In *Dissertation, ETH Zürich, Institut für Geodäsie und Photogrammetrie*, 1992.

[Mel97] A. Melling. Tracer particles and seeding for particle image velocimetry. *Meas. Sci. Technol.*, 8:1406–1416, 1997.

[Mic94] Z. Michaewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 1994.

[MS85] D. Mumford and J. Shah. Boundary detection by minimizing functionals, I. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 22–26, 1985.

[MS88] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and variational problems. *Comm. Pure Appl. Math.*, 42(5):577–685, 1988.

[OL00] K. Ohmi and H.-Y. Li. Particle-tracking velocimetry with new algorithms. *Meas. Sci. Technol.*, 11:603–616, 2000.

[ONI97] K. Ogawara, T. Noborizato, and S. Iida. A moving least square piv algorithm coupled with navier-stokes flow solver. In *Proc. 2nd Int. Workshop on PIV'97*, 1997.

[ONK00a] K. Okamoto, S. Nishio, and T. Kobayashi. Standard images for particle-image velocimetry. *Meas. Sci. Technol.*, 11:685–691, 2000.

[ONK+00b] K. Okamoto, S. Nishio, T. Kobayashi, T. Saga, and K. Takehara. Evaluation of the 3d-piv standard images (piv-std project). *Journal of Visualization*, 3(2):115–124, 2000.

[OSN00] T. Okuno, Y. Sugii, and S. Nishio. Image measurement of flow field using physics-based dynamic model. *Meas. Sci. Technol.*, 11:667–676, 2000.

[Par06] N. Paragios. Curve propagation, level set methods and grouping. In N. Paragios, Y. Chen, and O. Faugeras, editors, *Handbook of Mathematical Models in Computer Vision*, pages 144–159, 2006.

[PC05] E. G. Puckett and P. Colella. *Finite Difference Methods for Computational Fluid Dynamics (Cambridge Texts in Applied Mathematics)*. Cambridge University Press, 2005.

[Pra00] A.K. Prasad. Stereoscopic particle image velocimetry. *Exp. Fluids*, 29:103–116, 2000.

[PTVF92] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, UK, 2nd edition, 1992.

[QP98] G. M. Quénot and J. Pakleza. Particle image velocimetry with optical flow, 1998.

[Qué92] G. M. Quénot. The orthogonal algorithm for optical flow detection using dynamic programming. In *Proc. Intl.. Conf. on Acoustics, Speech and Signal Proc.*, pages 249–252, 1992.

[Qué99] G. M. Quénot. Performance evaluation of an optical flow technique applied to piv using the vsj standard images. In *Third International Workshop on PIV*, pages 579–584, 1999.

[RGS04] P. Ruhnau, C. Gütter, and C. Schnörr. A variational approach for particle tracking velocimetry. Comp. science series, technical report, Dept. Math. and Comp. Science, University of Mannheim, Germany, 2004.

[RGS05a] P. Ruhnau, C. Gütter, and C. Schnörr. A variational approach for particle tracking velocimetry. *Meas. Sci. Technol.*, 16(7):1449–1458, 2005.

[RGS05b] P. Ruhnau, C. Gütter, and C. Schnörr. Variational particle tracking velocimetry. In *Proc. 6th International Symposium on Particle Image Velocimetry, Pasadena, California, USA, September 21-23*, 2005.

[Rit08] W. Ritz. Über eine neue methode zur lösung gewisser variationsprobleme der mathematischen physik. *J. reine u. angew. Math.*, 135, 1908.

[RKNS04] P. Ruhnau, T. Kohlberger, H. Nobach, and C. Schnörr. Variational optical flow estimation for particle image velocimetry. In *Lasermethoden in der Strömungsmesstechnik*, volume 12, pages 30–1–30–8. Deutsche Gesellschaft für Laser-Anemometrie GALA e.V., 2004.

[RKNS05] P. Ruhnau, T. Kohlberger, H. Nobach, and C. Schnörr. Variational optical flow estimation for particle image velocimetry. *Exp. Fluids*, 38:21–32, 2005.

[RLK93] M. Raffel, B. Leitl, and J. Kompenhans. Data validation for particle image velocimetry. In *Laser Techniques and Applications in Fluid Mechanics*, pages 210–226. Springer, 1993.

[RS06a] P. Ruhnau and C. Schnörr. Optical stokes flow. In *13th International Symposium. Applications of Laser Techniques to Fluid Mechanics, Lisbon, Portugal*, 2006.

[RS06b] P. Ruhnau and C. Schnörr. Optical stokes flow: An imaging-based control approach. In *77th Annual Meeting of the Gesellschaft für Angewandte Mathematik und Mechanik e.V.*, 2006. In press.

[RS06c] P. Ruhnau and C. Schnörr. Optical stokes flow estimation: An imaging based control approach. *Exp. Fluids*, 2006. Accepted for publication.

[RSS06a] P. Ruhnau, A. Stahl, and C. Schnörr. On-line variational estimation of dynamical fluid flows with physics-based spatio-temporal regularization. In *Pattern Recognition, Proc. 26th DAGM Symposium*, 2006. In press.

[RSS06b] P. Ruhnau, A. Stahl, and C. Schnörr. On-line variational estimation of dynamical fluid flows with physics-based spatio-temporal regularization. *Meas. Sci. Technol.*, 2006. Submitted.

[RWK99] M. Raffel, C.E. Willert, and J. Kompenhans. *Particle Image Velocimetry. A Practical Guide.* Springer, 1999.

[RWK01] M. Raffel, C. Willert, and J. Kompenhans. *Particle Image Velocimetry*. Springer, 2001.

[RWW+96] M. Raffel, J. Westerweel, C. Willert, M. Gharib, and J. Kompenhans. Analytical and experimental investigations of dual-plane particle image velocimetry. *Optical Engineering*, 35:2067–2074, 1996.

[SBB04] A.M. Shinneeb, J.D. Bugg, and R Balachandar. Variable threshold outlier identification in piv data. *Meas. Sci. Technol.*, 15(9):1722–1732, 2004.

[Sca02] F. Scarano. Iterative image deformation methods in piv. *Meas. Sci. Technol.*, 13:R1–R19, 2002.

[Sch84] B.G. Schunck. The motion constraints equation for optical flow. In *Proceedings of the Seventh IEEE International Conference on Pattern Recognition*, pages 20–22, 1984.

[Sch91] C. Schnörr. Determining optical flow for irregular domains by minimizing quadratic functionals of a certain class. *International Journal of Computer Vision*, 6(1):25–38, 1991.

[Sch94] C. Schnörr. Segmentation of visual motion by minimizing convex non-quadratic functionals. In *12th Int. Conf. on Pattern Recognition, Jerusalem, Israel*, volume A, pages 661–663. IEEE Computer Society Press, 1994.

[Sch99] C. Schnörr. Variational methods for adaptive image smoothing and segmentation. In B. Jähne, H. Haussecker, and P. Geißler, editors, *Handbook on Computer Vision and Applications: Signal Processing and Pattern Recognition*, volume 2, pages 451–484, 1999.

[SDG02] H. Spies, T. Dierig, and C. Garbe. Local models for dynamic processes in image sequences. In *Workshop Dynamic Perception*, pages 59–64, Bochum, Germany, November 2002.

[Sim93] E. P. Simoncelli. *Distributed representation and analysis of visual motion*. PhD thesis, Massachusetts Institute of Technology, 1993.

[SM98] J. Sheng and H. Meng. A genetic algorithm particle pairing technique for 3d velocity field extraction in holographic particle image velocimetry. *Exp. Fluids*, 25(5–6):461–473, 1998.

[SR99] F. Scarano and M.L. Riethmuller. Iterative multigrid approach in piv image processing with discrete window offset. *Exp. Fluids*, 26:513–523, 1999.

[SR01] A. Stitou and M. L. Riethmuller. Extension of piv to super resolution using ptv. *Meas. Sci. Technol.*, 12:1398–1403, 2001.

[SRS06] A. Stahl, P. Ruhnau, and C. Schnörr. A distributed-parameter approach to dynamic image motion. In *Int. Workshop on The Representation and Use of Prior Knowledge in Vision*, ECCV 2006. to appear.

[SSKH00] T. Saga, S. Segawa, T. Kobayashi, and H. Hu. Development and evaluation of an improved correlation based ptv method. In *Proc. 6. Triennial Symposium on Fluid Control, Measurement and Visualization, Sherbrocke, Canada*, 2000.

[Sut93] D. Suter. Mixed finite elements and whitney forms in visual reconstruction. In *Geometric Methods in Computer Vision II*, pages 51–62, 1993.

[Sut94a] D. Suter. Mixed-finite element based motion estimation. *Innov. Tech. Biol. Med.*, 15(3), 1994.

[Sut94b] D. Suter. Motion estimation and vector splines. In *Proc. Conference on Computer Vision and Pattern Recognition*, pages 939–942, 1994.

[TD95] P.T. Tokumaru and P.E. Dimotakis. Image correlation velocimetry. *Exp. Fluids*, 19:1–15, 1995.

[UYO89] T. Uemura, F. Yamamoto, and K. Ohmi. A high speed algorithm of image analysis for real time measurement of two-dimensional velocity distribution. *Flow Visualization*, pages 129–133, 1989.

[Ver84] R. Verürth. Error estimation for mixed finite element approximation of the stokes equation. *RAIRO Anal. Numer. Analysis*, 18:175–182, 1984.

[WALL97] R.P. Wildes, M.J. Amabile, A.-M. Lanzillotto, and T.-S. Leu. Physically based fluid flow recovery from image sequences. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, page 969, Washington, DC, USA, 1997. IEEE Computer Society.

[WBL03] C. H. Westergaard, M. Brede, and A. Leder. Time-space analysis of time resolved piv data. In *Proc. 5th Int. Symp. on Particle Image Velocimetry (PIV'03),Busan, Korea*, 2003.

[WDG97] J. Westerweel, D. Dabiri, and M. Gharib. The effect of a discrete window offset on the accuracy of cross-correlation analysis of digital piv recordings. *Exp. Fluids*, 23:20–28, 1997.

[Wes93] J. Westerweel. *Digital particle image velocimetry*. PhD thesis, Delft University, 1993.

[Wes94] J. Westerweel. Efficient detection of spurious vector in particle image velocimetry data. *Exp. Fluids*, pages 236–247, 1994.

[WS01a] J. Weickert and C. Schnörr. A theoretical framework for convex regularizers in pde-based computation of image motion. *Int. J. Computer Vision*, 45(3):245–264, 2001.

[WS01b] J. Weickert and C. Schnörr. Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision*, 14(3):245–255, 2001.

[WS05] J. Westerweel and F. Scarano. Universal outlier detection for piv data. *Exp. Fluids*, 39(6):1096–1100, 2005.

[YRMS05] J. Yuan, P. Ruhnau, E. Mémin, and C. Schnörr. Discrete orthogonal decomposition and variational fluid flow estimation. In *5th International Conference on Scale Space and PDE Methods in Computer Vision, Hofgeismar, Germany*, 2005.